

For Reference

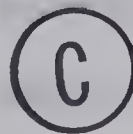
NOT TO BE TAKEN FROM THIS ROOM

Ex libris
UNIVERSITATIS
ALBERTAEENSIS



THE UNIVERSITY OF ALBERTA
NOISE REDUCTION IN DEADBEAT SYSTEMS

by



DONALD HENRY KJOSNESS

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE
OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA

FALL, 1971

Thesis
1971 F
40 D

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read,
and recommend to the Faculty of Graduate Studies for
acceptance, a thesis entitled "Noise Reduction in
Deadbeat Systems", submitted by Donald Henry Kjosness
in partial fulfilment of the requirements for the degree
of Doctor of Philosophy.

ABSTRACT

This thesis presents a design criterion for calculating non-minimum time deadbeat digital controllers, for a unit step input, which minimize the noise transfer of a sampled data system when white noise is present. The system consisted of a linear plant, a digital controller, a zero-order hold and a unity feedback branch.

The non-minimum time deadbeat controllers are used in place of the minimum time deadbeat controllers to improve the noise performance of the system. All deadbeat requirements are met by the non-minimum time deadbeat controllers except that the error for a unit step input is not reduced to zero in minimum time.

The design criterion is practical in application as it depends only on the plant parameters and system sampling rate. No explicit reference to the noise input is contained in the criterion.

The non-minimum time controllers may have any number of extra terms as the noise reduction increases with each extra term. Beyond a certain number of extra terms, the noise reduction becomes uneconomical because of the extra controller complexity and slower system response time. The length of this extension varies depending on the design requirements of the system. In this thesis, controllers with up to three extra terms were considered.

The non-minimum time controllers were tested using both a hybrid and a purely digital simulation to verify the minimum noise transfer requirement.

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Professor Y. J. Kingma for his assistance and cooperation during the research and preparation of this thesis. The author also wishes to thank the other staff members and graduate students in the Department of Electrical Engineering for their assistance.

The financial support supplied by the National Research Council and the Department of Electrical Engineering is gratefully acknowledged.

The author wishes to thank his parents and brother and sister for their encouragement and interest.

TABLE OF CONTENTS

	PAGE
INTRODUCTION	1
CHAPTER I GENERAL SYSTEM CONSIDERATIONS	5
1-1 System Configuration	5
1-2 Types of Plants Considered	6
1-3 \mathcal{Z} - Transforms of Plants Considered	8
1-4 The Form of The Digital Controllers	12
1-5 State Equations and Flow Graphs For $G_1(s)=1/s(s+\alpha)$	16
1-6 State Equations and Flow Graphs For $G_2(s)=1/s(s+\alpha)(s+\beta)$	18
1-7 State Equations and Flow Graphs For $G_3(s)=(s+\beta)/s(s+\alpha)$	19
1-8 State Equations and Flow Graphs For $G_4(s)=1/s(s^2+2\zeta\omega_n s+\omega_n^2)$	20
CHAPTER II HYBRID COMPUTER SIMULATION	24
2-1 Hybrid Computer System	25
2-2 Hybrid System Simulation	26
2-3 State Variable Method For Calculating Minimum and Extra Term Controllers	30
2-4 z-Domain Method For Calculating Minimum and Extra Term Controllers	42
2-5 Noise Source Used On Hybrid Simulation	49
2-6 Test Procedure On The Hybrid Simulation	51

		PAGE
CHAPTER III	HYBRID SIMULATION RESULTS AND THE DERIVATION OF A DESIGN CRITERION	53
3-1	Systems Tested by Hybrid Simulation	53
3-2	Graphical Results and Summary	54
3-3	Considerations Involved In The Development of A Design Criterion For An Extended Controller	61
3-4	Criteria Considered	62
3-5	Derivation of The Design Criterion	64
3-6	Comparison of Criterion Results and Hybrid Results	71
CHAPTER IV	DIGITAL COMPUTER SIMULATION	73
4-1	Method of Digital Simulation	74
4-2	Digital Computer Simulation Test Procedure	76
4-3	Noise Sources Simulated By The Random Number Generators	78
CHAPTER V	CONTROLLERS EXTENDED ONE TERM	83
5-1	Algorithm Used In The Digital Simulation to Calculate The Predicted PlCs	84
5-2	Results For The PlCs of $G_1(s) = 1/s(s+\alpha)$	92
5-3	Results For The PlCs of $G_2(s) = 1/s(s+\alpha)(s+\beta)$	98
5-4	Results For The PlCs of $G_3(s) = (s+\beta)/s(s+\alpha)$	102
5-5	Results For The PlCs of $G_4(s) = 1/s(s^2 + 2\zeta\omega_n s + \omega_n^2)$	106

	PAGE
5-6	Results For The Four Plants When A Ramp or An Acceleration Input Is Used 109
5-7	Some Conclusions For The PlCs 113
CHAPTER VI	CONTROLLERS EXTENDED TWO TERMS 117
6-1	Algorithm Used In The Digital Simulation To Calculate The Predicted P2Cs 117
6-2	Results For P2Cs of $G_1(s) = 1/s(s+\alpha)$ 126
6-3	Results For P2Cs of $G_2(s) = 1/s(s+\alpha)(s+\beta)$ 129
6-4	Results For P2Cs of $G_3(s) = (s+\beta)/s(s+\alpha)$ 133
6-5	Results For P2Cs of $G_4(s) = 1/s(s^2+2\zeta\omega_n s+\omega_n^2)$ 133
6-6	Some Conclusions For The P2Cs 139
CHAPTER VII	OTHER EXTENDED CONTROLLERS 142
7-1	Predicted P3Cs of $G_1(s)$ 142
7-2	PlCs With A Reduced Sampling Period 144
7-3	Straight Line Approximations To The Predicted PlCs 149
CHAPTER VIII	GENERAL CONCLUSIONS 154
8-1	Conclusions And Observations 154
8-2	Extensions And Discussion 156
BIBLIOGRAPHY	158

		PAGE
APPENDIX A	HYBRID COMPUTER SIMULATION	160
A-1	Digital Controller Realization	160
A-2	Digital Program Used In Hybrid Simulation	166
APPENDIX B	DIGITAL COMPUTER SIMULATION OF THE DIGITAL CONTROLLERS	168
B-1	Derivation of Program	168
B-2	Digital Program	169

LIST OF TABLES

		PAGE
TABLE 1.1	TRANSFER FUNCTIONS OF ALL PLANTS IN s AND z PLANES	13
TABLE 2.1(a)	COEFFICIENTS OF MINIMUM DEADBEAT CONTROLLER FOR $G_1(s) = 1/s(s+\alpha)$	40
TABLE 2.1(b)	COEFFICIENTS OF 3 TERM DEADBEAT CONTROLLER FOR $G_1(s) = 1/s(s+\alpha)$ (ONE EXTRA TERM)	41
TABLE 2.2(a)	COEFFICIENTS OF MINIMUM DEADBEAT CONTROLLER FOR $G_2(s) = 1/s(s+\alpha)$ (s+β)	48
TABLE 2.2(b)	COEFFICIENTS OF 4 TERM DEADBEAT CONTROLLER FOR $G_2(s) = 1/s(s+\alpha)$ (s+β) (ONE EXTRA TERM)	48
TABLE 3.1	NOISE TRANSFERS AND CONFIDENCE LIMITS FOR SYSTEMS CONTAINING THE MINIMUM TIME CONTROLLERS MEASURED ON THE HYBRID	55
TABLE 3.2	SUMMARY OF HYBRID COMPUTER RESULTS	59
TABLE 3.3	COEFFICIENTS OF $BE_1(mT)$ FOR THE VARIOUS CYCLES OF THE DIGITAL CONTROLLER	60
TABLE 3.4	COMPARISON OF HYBRID AND NCE RESULTS	72
TABLE 4.1	FOURIER COEFFICIENT STATISTICS FOR RANDOM NUMBER NOISE SOURCES	81
TABLE 5.1(a)	COEFFICIENTS OF EQUATIONS (5.1) TO (5.6) FOR THE PLCs OF $G_1(s)$ AND $G_2(s)$	89

TABLE 5.1(b)	COEFFICIENTS OF EQUATIONS (5.1) TO (5.6) FOR THE PLCs OF $G_3(s)$ AND $G_4(s)$	90
TABLE 5.2	COMPARISON OF PREDICTED AND GRID SEARCHED PLCs FOR $G_1(s)$ WITH $T=1$ SECOND	98
TABLE 6.1(a)	COEFFICIENTS OF THE P2C FOR CALCULATION AS FUNCTIONS OF e_1 AND e_2 FOR $G_1(s)$ AND $G_2(s)$	124
TABLE 6.1(b)	COEFFICIENTS OF THE P2C FOR CALCULATION AS FUNCTIONS OF e_1 AND e_2 FOR $G_3(s)$ AND $G_4(s)$	125

LIST OF FIGURES

		PAGE
FIGURE 1.1	BASIC SYSTEM CONFIGURATION WITH A UNIT STEP INPUT PLUS NOISE	7
FIGURE 1.2	DEADBEAT RESPONSE TO A UNIT STEP INPUT	7
FIGURE 1.3	DIRECT DIGITAL PROGRAMMING REALIZATION OF $D(z)$	15
FIGURE 1.4	FLOW GRAPH FOR SYSTEM CONTAINING $G_1(s)$	22
FIGURE 1.5	FLOW GRAPH FOR SYSTEM CONTAINING $G_2(s)$	22
FIGURE 1.6	FLOW GRAPH FOR SYSTEM CONTAINING $G_3(s)$	23
FIGURE 1.7	FLOW GRAPH FOR SYSTEM CONTAINING $G_4(s)$	23
FIGURE 2.1	TEST SYSTEM TO MEASURE SYSTEM ERROR WHEN NOISE PRESENT	28
FIGURE 2.2	TEST SYSTEM IMPLEMENTED ON HYBRID TO MEASURE SYSTEM ERROR WHEN NOISE PRESENT	28
FIGURE 2.3	BLOCK DIAGRAM OF HYBRID COMPUTER SIMULATION	29
FIGURE 2.4	NOISE SPECTRUM OF SERVOMEX R. G. 77 RANDOM NOISE GENERATOR	50
FIGURE 2.5	AUTO-SPECTRAL DENSITY OF BANDLIMITED WHITE NOISE AFTER SAMPLING	50
FIGURE 3.1	HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 1)$ WITH $T=1$ SEC.	56
FIGURE 3.2	HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 3)$ WITH $T=1$ SEC.	57
FIGURE 3.3	HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 1)$ WITH $T=0.5$ SEC.	57

	PAGE
FIGURE 3.4	HYBRID RESULTS FOR 4 TERM CONTROLLERS OF $G_2(s) = 1/s(s + 1)(s + 2)$ WITH T=1 SEC.
	58
FIGURE 3.5	HYBRID RESULTS FOR 4 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 1)$ WITH T=1 SEC. AND $e_2=0.5$
	58
FIGURE 4.1	SIMPLIFIED BLOCK DIAGRAM OF DIGITAL COMPUTER SIMULATION
	77
FIGURE 5.1	PREDICTED e_1 FOR P1Cs OF $G_1(s)$ WITH T=1 SEC.
	94
FIGURE 5.2	NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs OF $G_1(s)$ WITH T=1 SEC. USING GAUSS2 AND RANDUA
	95
FIGURE 5.3	PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2A AND RANDUA, FOR P1Cs OF $G_1(s)$ WITH T=1 SEC.
	95
FIGURE 5.4	95% CONFIDENCE LIMITS ON MEAN NORMALIZED NOISE TRANSFER USING P1Cs OF $G_1(s)$ WITH T=1 SEC.
	97
FIGURE 5.5	95% CONFIDENCE LIMITS ON MEAN VALUES OF GRID SEARCHED e_1 COMPARED TO PREDICTED e_1 FOR P1Cs OF $G_1(s)$ WITH T=1 SEC.
	97
FIGURE 5.6	PREDICTED e_1 FOR P1Cs OF $G_2(s)$ WITH T=1 SEC.
	100
FIGURE 5.7	PREDICTED e_1 FOR P1Cs OF $G_2(s)$ WITH T=1 & 10 SEC.
	100
FIGURE 5.8	NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs OF $G_2(s)$ WITH T=1 SEC. USING GAUSS2A
	101

	PAGE
FIGURE 5.9	PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2, FOR PLCs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.
	101
FIGURE 5.10	PREDICTED e_1 FOR PLCs OF $G_3(s)$ WITH $T=1$ SEC.
	103
FIGURE 5.11	NORMALIZED NOISE TRANSFERS USING PREDICTED PLCs OF $G_3(s)$ WITH $T=1$ SEC. USING GAUSS2
	104
FIGURE 5.12	PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2, FOR PLCs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.
	104
FIGURE 5.13	MAXIMUM OVERSHOOTS TO A STEP INPUT FOR PREDICTED PLCs AND MINIMUM TIME DEADBEAT CONTROLLERS OF $G_2(s)$ $T=1$ SEC.
	105
FIGURE 5.14	PREDICTED e_1 FOR PLCs OF $G_4(s)$ WITH $T=1$ SEC.
	107
FIGURE 5.15	NORMALIZED NOISE TRANSFERS USING PREDICTED PLCs OF $G_4(s)$ WITH $T=1$ SEC. USING GAUSS2
	108
FIGURE 5.16	PREDICTED e_1 FOR PLCs OF $G_4(s)$ WITH $T=5$ SEC.
	108
FIGURE 5.17	NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED PLCs OF $G_1(s)$ WITH $T=1$ SEC.
	111
FIGURE 5.18	NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED PLCs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.
	111
FIGURE 5.19	NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED PLCs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.
	112

	PAGE
FIGURE 5.20	NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED P1Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC. 112
FIGURE 6.1	PREDICTED e_1 AND e_2 FOR P2Cs OF $G_1(s)$ WITH $T=1$ SEC. 127
FIGURE 6.2	NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2 127
FIGURE 6.3	PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_1(s)$ WITH $T=1$ SEC. 128
FIGURE 6.4	NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2 128
FIGURE 6.5	PREDICTED e_1 AND e_2 FOR P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. 131
FIGURE 6.6	NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. USING GAUSS2 131
FIGURE 6.7	PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. 132
FIGURE 6.8	NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. USING GAUSS2 132
FIGURE 6.9	PREDICTED e_1 AND e_2 FOR P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC. 134
FIGURE 6.10	NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC. 134

FIGURE 6.11	PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.	135
FIGURE 6.12	NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC. USING GAUSS2	135
FIGURE 6.13	PREDICTED e_1 AND e_2 FOR P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC.	136
FIGURE 6.14	NORMALIZED NOISE TRANSFERS USING PREDICTED PlCs AND P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC. USING GAUSS2	136
FIGURE 6.15	PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC.	137
FIGURE 6.16	NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC. USING GAUSS2	137
FIGURE 7.1	PREDICTED e_1 , e_2 AND e_3 FOR P3Cs OF $G_1(s)$ WITH $T=1$ SEC.	145
FIGURE 7.2	NORMALIZED NOISE TRANSFERS USING PREDICTED PlCs, P2Cs AND P3Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2	145
FIGURE 7.3	NORMALIZED NOISE TRANSFERS USING PREDICTED PlCs OF $G_1(s)$ WITH $T=2/3$ & 1 SEC. USING GAUSS2AM	148
FIGURE 7.4	NORMALIZED NOISE TRANSFERS USING PREDICTED PlCs OF $G_2(s)$ WITH $\beta=1$ AND $T=3/4$ & 1 SEC. USING GAUSS2AM	148

FIGURE 7.5	PREDICTED e_1 AND STRAIGHT LINE APPROXIMATION e_1 FOR PLCs OF $G_1(s)$ WITH $T=1$ SEC.	152
FIGURE 7.6	PREDICTED e_1 AND STRAIGHT LINE APPROXIMATION e_1 FOR PLCs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.	152
FIGURE A.1	HYBRID REALIZATION OF THE DIGITAL CONTROLLER	165

INTRODUCTION

When sampled data control systems containing a linear plant with a digital controller to give deadbeat response to a unit step input are subjected to any noise input, the errors which occur are rather large. This is caused by the high gains required in the digital controller to satisfy the deadbeat requirements for a unit step input.

These requirements are:

1. The system reaches zero steady-state error after a minimum number of sampling periods with no intersampling ripple.
2. The system normally should have no transient overshoots. In some cases this overshoot is unavoidable. Then the transient response should be as fast as possible, with a finite settling time measured at the sampling instants.

The controller which satisfies these requirements for a given system is unique. Therefore if any alterations are made to the controller to improve the noise performance of the system, the resulting controller cannot satisfy all deadbeat requirements.

In this thesis, a design criterion is presented which yields controllers that minimize the noise transfer of the system in the presence of noise which is assumed to be white.

These controllers are designed to satisfy all deadbeat requirements except the minimum response time restriction. Therefore the predicted controllers may be called "non-minimum time deadbeat digital controllers."

When using this method, the plant parameters and the sampling rate of the system are normally held constant. Thus the non-minimum time deadbeat controller may be directly substituted for the corresponding minimum time deadbeat controller for a given system to achieve the maximum noise reduction.

This method of extending the controller to improve the noise performance of the system has the following advantages:

1. The gains of the controller are lowered and thus the level of the transient response is lowered.
2. Because the controller is no longer a minimum time controller, extra degrees of freedom are available in its design. These may be used to minimize the transfer of noise by the system.

In effect, the predicted non-minimum time controllers are designed to do as little as possible to the signal passing through them. This is contrary to the more standard approach. Usually the system loop gain is made as large as possible to allow very rapid correction of any disturbances. This is generally accomplished by the insertion of a network in the feedback path to raise the loop gain without altering

the forward path.

It would be possible to insert a feedback compensator in the deadbeat system to improve the noise performance. However it is a more complicated problem in terms of the system realization than is the simple addition of terms to the controller.

Authors such as Tou^{1,8,9}, Ackermann² and Chang¹⁴ have published methods for calculating controllers which minimize the noise transfer of a sampled data system. However these methods do not readily lend themselves to the case when deadbeat is an additional constraint. Some of the reasons for this observation are discussed in Chapter 3.

To the best of the author's knowledge, no published literature contains any specific reference to the noise reduction problem for deadbeat systems. Therefore the design criterion presented in this thesis is derived independently of published methods.

The proposed criterion is stated in Chapter 3. This criterion may be used to calculate controllers which have any number of extra terms. However at some point, the additional noise reduction becomes uneconomic when compared to the additional controller complexity required.

The criterion presented is practical in application and it is expressed only in terms of the coefficients of the non-minimum time controller. These controller coefficients are functions of the plant parameters and the sampling rate of the system. The noise input does not appear explicitly

in the criterion. To allow the design criterion to take this form, pure white noise must be assumed as it has no cross-correlation with a unit step input. Therefore the two signals may be assumed to be totally independent.

In Chapters 1 and 2, the form of the system considered is presented along with other general information. The details of a hybrid computer simulation used for the initial tests of the non-minimum time controllers are presented in Chapter 3. In Chapter 4, the details of a purely digital test simulation used to obtain the results presented in Chapters 5, 6 and 7 are given.

The results presented in Chapters 5, 6 and 7 are for non-minimum time controllers with one, two or three extra terms respectively. In Chapter 5, there is no noise input to the system when the ramp and acceleration inputs are used.

In Chapter 7, results are presented for the case when the sampling rate of the system is not held constant. The sampling rates used are such that the total response times to deadbeat for both the minimum time controllers and the predicted non-minimum time controllers with one extra term are the same. This approach overcomes the disadvantage of the slow response of the non-minimum time controller while still providing some noise reduction.

CHAPTER I

GENERAL SYSTEM CONSIDERATIONS

Throughout this thesis, some basic information on the systems tested will be required. For each plant to be considered, the following information is presented in this chapter:

1. The transfer function of the plant.
2. The z - transform of the plant.
3. The discrete signal flow graph for the system.
4. The discrete state equations of the system.

1-1 SYSTEM CONFIGURATION

The basic system considered is a unity feedback system with a noise contaminated input or feedback signal as shown in FIG. 1.1.

In this system, the noise is shown as injected in the feedback path. Since this is a unity feedback system, the noise could actually be contaminating the input signal or be introduced by the feedback branch.

The general form of the digital controller is:

$$D(z) = \frac{h_0 + h_1 z^{-1} + h_2 z^{-2} + \dots + h_n z^{-n}}{1 + e_1 z^{-1} + e_2 z^{-2} + \dots + e_n z^{-n}} \quad (1.1)$$

where n is the total number of sampling periods required to reach deadbeat for a unit step input. The zero order hold has the transfer function:

$$G_z(s) = \frac{1 - e^{-sT}}{s} \quad (1.2)$$

where T is the sampling rate of the system. For most calculations, the zero order hold and the plant are combined as follows:

$$G_{pm}(s) = G_z(s) G_m(s) \quad (1.3)$$

where $G_m(s)$ is the particular plant under consideration.

1-2 TYPES OF PLANTS CONSIDERED

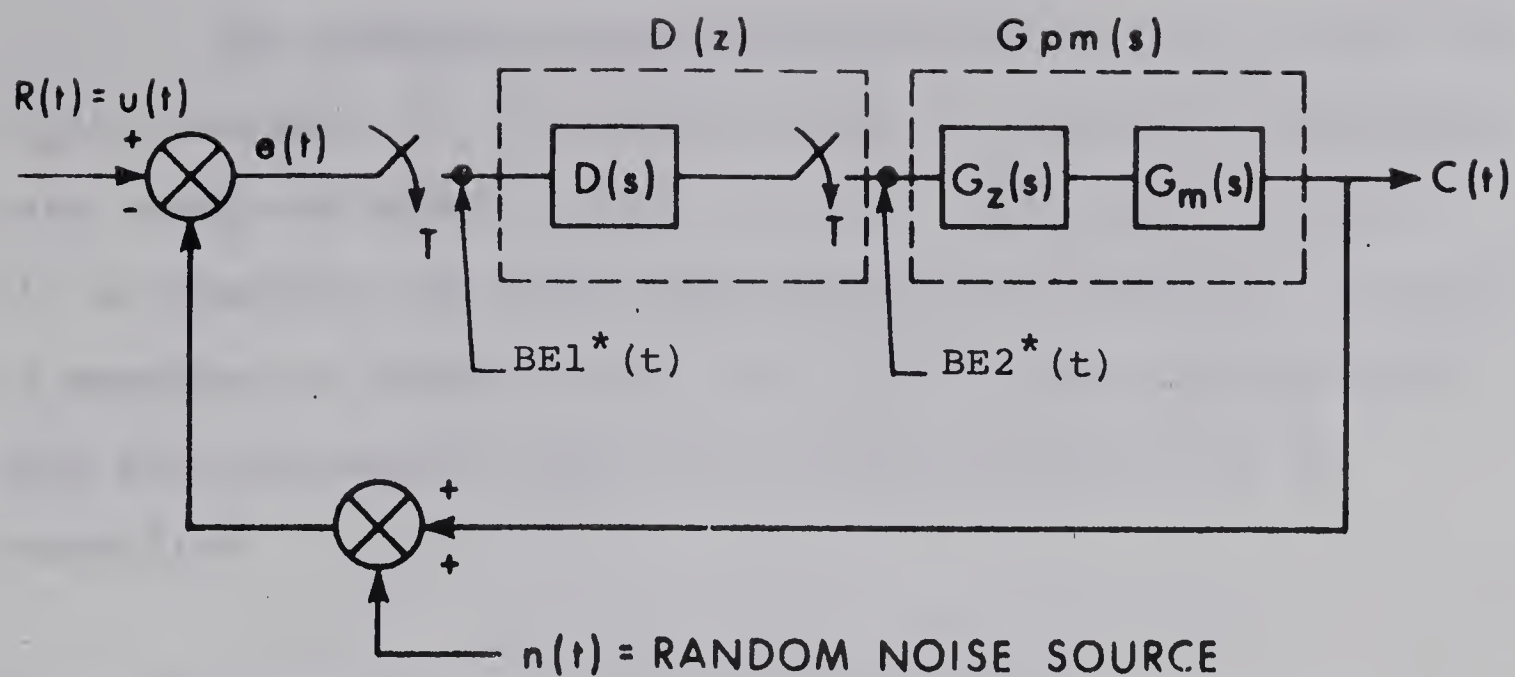
There are several basic plant configurations which represent the majority of system classes possible. Four such plants are considered. These are:

$$G_1(s) = 1/s (s + \alpha) \quad (1.4)$$

$$G_2(s) = 1/s (s + \alpha) (s + \beta) \quad (1.5)$$

$$G_3(s) = (s + \alpha) / s (s + \beta) \quad (1.6)$$

$$G_4(s) = 1/s (s^2 + 2\zeta \omega_n s + \omega_n^2) \quad (1.7)$$



$G_z(s) = \text{ZERO ORDER HOLD}$

$G_m(s) = \text{PLANT UNDER CONSIDERATION}$

FIGURE 1.1 BASIC SYSTEM CONFIGURATION WITH A UNIT STEP INPUT PLUS NOISE

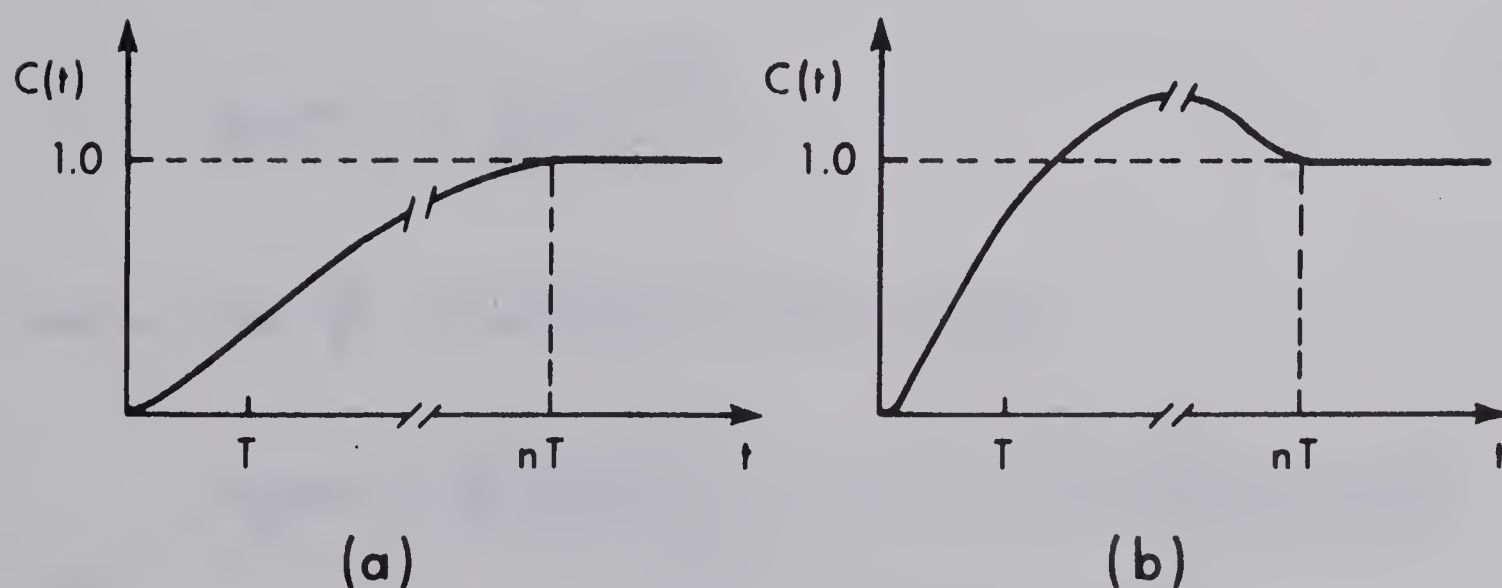


FIGURE 1.2 DEADBEAT RESPONSE TO A UNIT STEP INPUT

The deadbeat responses exhibited by $G_1(s)$, $G_2(s)$ and $G_4(s)$, whether for the minimum time or extended controllers, are always as shown in FIG. 1.2 (a). However for $G_3(s)$, it is possible for some combinations of parameters to yield a response as shown in FIG. 1.2 (b).³ This response still has no intersample ripple and reaches unity at nT as specified.

1-3 \mathcal{Z} - TRANSFORMS OF PLANTS CONSIDERED

As shown in (1.3), the plant $G_m(s)$ and the zero-order hold $G_z(s)$ are combined as $G_{pm}(s)$. For later use, it is necessary to calculate $G_{pl}(z)$. This section contains the calculation of $G_{pl}(z)$. The transforms for the other three plants are then listed.

Substituting (1.2) and (1.4) into (1.3) yields:

$$G_{pl}(s) = \frac{1 - e^{-sT}}{s^2 (s + \alpha)} \quad (1.8)$$

Taking the \mathcal{Z} - Transform of (1.8) yields:

$$G_{pl}(z) = \mathcal{Z} \left[\frac{1 - e^{-sT}}{s^2 (s + \alpha)} \right] = (1 - z^{-1}) \mathcal{Z} \left[\frac{1}{s^2 (s + \alpha)} \right]$$

$$G_{pl}(z) = \left(\frac{1 - z^{-1}}{\alpha} \right) \left[\frac{T z}{(z-1)^2} - \frac{(1 - e^{-\alpha T}) z}{\alpha (z-1) (z - e^{-\alpha T})} \right] \quad (1.9)$$

From (1.9), define the following:

$$A = e^{-\alpha T} \quad (1.10)$$

Substituting (1.10) into (1.9) yields the simplified expression:

$$G_{p1}(z) = \frac{z (\alpha T + A - 1) + (1 - A - \alpha T A)}{\alpha^2 (z - 1) (z - A)} \quad (1.11)$$

To simplify (1.11), define the following:

$$R1 = (\alpha T + A - 1) / \alpha^2 \quad (1.12)$$

$$R2 = (1 - A - \alpha T A) / \alpha^2 \quad (1.13)$$

Substituting (1.12) and (1.13) into (1.11), yields the z -transform of $G_{p1}(s)$ as:

$$G_{p1}(z) = \frac{(R1) z + R2}{(z - 1) (z - A)} \quad (1.14)$$

The z -transforms for the remaining three plants are derived in a similar fashion.

Thus for $G_{p2}(s)$, the z -transform is:

$$G_{p2}(z) = \frac{(R7) z^2 + (R8) z + U}{(z - 1) (z - A) (z - B)} \quad (1.15)$$

where

$$A = e^{-\alpha T} \quad (1.16)$$

$$B = e^{-\beta T} \quad (1.17)$$

$$R1 = 1 / \alpha \beta \quad (1.18)$$

$$R2 = - (1 / \alpha^2 \beta + 1 / \alpha \beta^2) \quad (1.19)$$

$$R3 = 1 / \alpha^2 (\beta - \alpha) \quad (1.20)$$

$$R4 = 1 / \beta^2 (\alpha - \beta) \quad (1.21)$$

$$R7 = T (R1) - R2 (1 + A + B) - R3 (2 + B) - R4 (2 + A) \quad (1.22)$$

$$R8 = - (R1) T (A + B) + R2 (A + B + AB) + R3 (1 + 2B) + R4 (1 + 2A) \quad (1.23)$$

$$U = R1 (TAB) - R2 (AB) - (R3) B - (R4) A \quad (1.24)$$

For the plant, $G_{p3}(s)$, the \mathcal{Z} - transform is:

$$G_{p3}(z) = \frac{(B) z + C}{(z - 1) (z - A)} \quad (1.25)$$

where

$$A = e^{-\alpha T} \quad (1.26)$$

$$B = (\beta (\alpha T + A - 1) + \alpha (1 - A)) / \alpha^2 \quad (1.27)$$

$$C = (\beta (1 - A - A\alpha T) - \alpha (1 - A)) / \alpha^2 \quad (1.28)$$

Finally, the \mathcal{Z} - transform of $G_{p4}(s)$ is:

$$G_{p4}(z) = \frac{(A1) z^2 + (A2) z + A3}{(z - 1) (z^2 + (AC8) z + AC9)} \quad (1.29)$$

where

$$A = \zeta \omega_n \quad (1.30)$$

$$B = \omega_n (1 - \zeta^2)^{\frac{1}{2}} \quad (1.31)$$

$$AC1 = 1 / (A^2 + B^2) \quad (1.32)$$

$$AC2 = -2 (A) (AC1)^2 \quad (1.33)$$

$$AC3 = -AC2 \quad (1.34)$$

$$AC4 = -AC1 - 2 (A) AC2 \quad (1.35)$$

$$AC5 = T (AC1) \quad (1.36)$$

$$AET = \text{EXP} (-AT) \quad (1.37)$$

$$BT = B (T) \quad (1.38)$$

$$AC6 = AC2 (AET) \text{COS} (BT) \quad (1.39)$$

$$AC7 = ((2 (A^2) AC1 - 1) AC1 (AET) \text{SIN} (BT)) / B \quad (1.40)$$

$$AC8 = -2(AET) \cos(BT) \quad (1.41)$$

$$AC9 = (AET)^2 \quad (1.42)$$

$$A1 = AC5 + AC2(AC8 + 1) + AC6 + AC7 \quad (1.43)$$

$$A2 = AC5(AC8) + AC2(AC9 - AC8) + AC3 - 2(AC6 + AC7) \quad (1.44)$$

$$A3 = AC9(AC5 - AC2) + AC6 + AC7 \quad (1.45)$$

Refer to Table 1.1 for the listing of $G_m(s)$, $G_{pm}(s)$ and $G_{pm}(z)$ for all cases.

1-4 THE FORM OF THE DIGITAL CONTROLLER

The pulse transfer function $D(z)$ of the digital controller is of the form:

$$D(z) = \frac{BE2(z)}{BE1(z)} = \frac{h_0 + h_1 z^{-1} + \dots + h_n z^{-n}}{1 + e_1 z^{-1} + \dots + e_n z^{-n}} \quad (1.46)$$

where $BE1(z)$ and $BE2(z)$ represent the \mathcal{Z} -transforms of the input and output signals of the controller as shown in FIG. 1.1. Cross-multiplying in (1.46) and taking the inverse \mathcal{Z} -transform yields the simplified form:⁴

$$BE2^*(t) + \sum_{k=1}^n (e_k) BE2^*(t - kT) = \sum_{k=0}^n (h_k) BE1^*(t - kT) \quad (1.47a)$$

$\frac{G_m(s)}{G_{pm}(s)}$	$\frac{G_{pm}(s)}{G_{pm}(z)}$
$G_1(s) = \frac{1}{s(s+\alpha)}$	$G_{p1}(z) = \frac{(R1)z + R2}{(z-1)(z-A)}$
$G_2(s) = \frac{1}{s(s+\alpha)(s+\beta)}$	$G_{p2}(z) = \frac{(R7)z^2 + (R8)z + U}{(z-1)(z-A)(z-B)}$
$G_3(s) = \frac{(s + \beta)}{s(s + \alpha)}$	$G_{p3}(z) = \frac{(B)z + C}{(z-1)(z-A)}$
$G_4(s) = \frac{1}{(s^2 + 2\zeta\omega_n s + \omega_n^2)}$	$G_{p4}(z) = \frac{(A1)z^2 + (A2)z + A3}{(z-1)(z^2 + (AC8)z + AC9)}$

TABLE 1.1 THE TRANSFER FUNCTIONS OF ALL PLANTS IN s AND z PLANES

Rearrange (1.47a) as follows:

$$\begin{aligned} \text{BE2}^*(t) &= \sum_{k=0}^n (h_k) \text{BE1}^*(t - kT) - \\ &\quad \sum_{k=1}^n (e_k) \text{BE2}^*(t - kT) \end{aligned} \quad (1.47b)$$

From (1.47b), define the following:

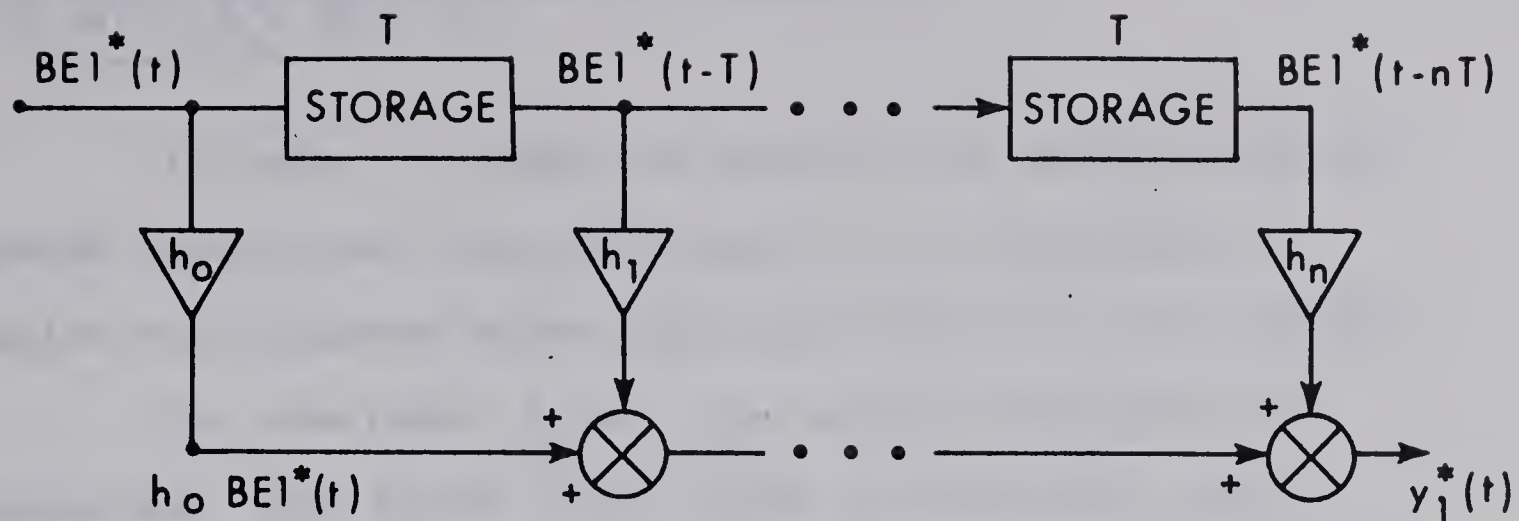
$$y_1^*(t) = \sum_{k=0}^n (h_k) \text{BE1}^*(t - kT) \quad (1.48)$$

$$y_2^*(t) = \sum_{k=1}^n (e_k) \text{BE2}^*(t - kT) \quad (1.49)$$

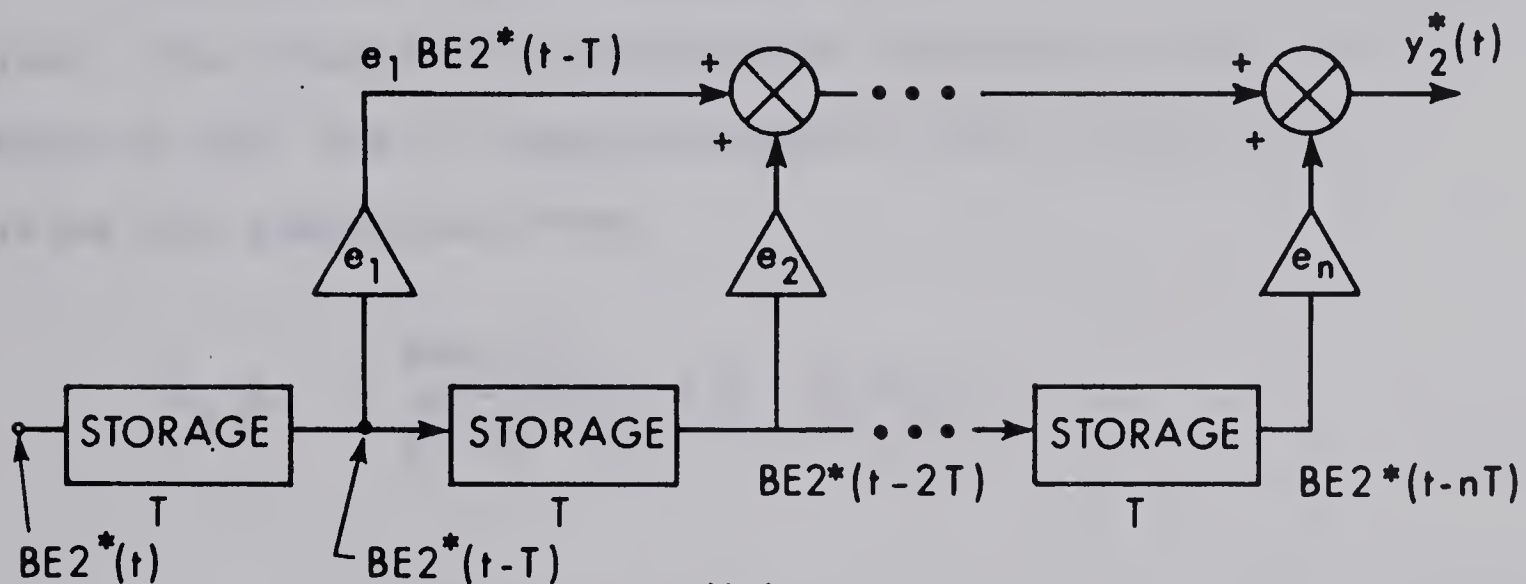
Rewriting (1.47b) using (1.48) and (1.49) yields the simplified equation:

$$\text{BE2}^*(t) = y_1^*(t) - y_2^*(t) \quad (1.50)$$

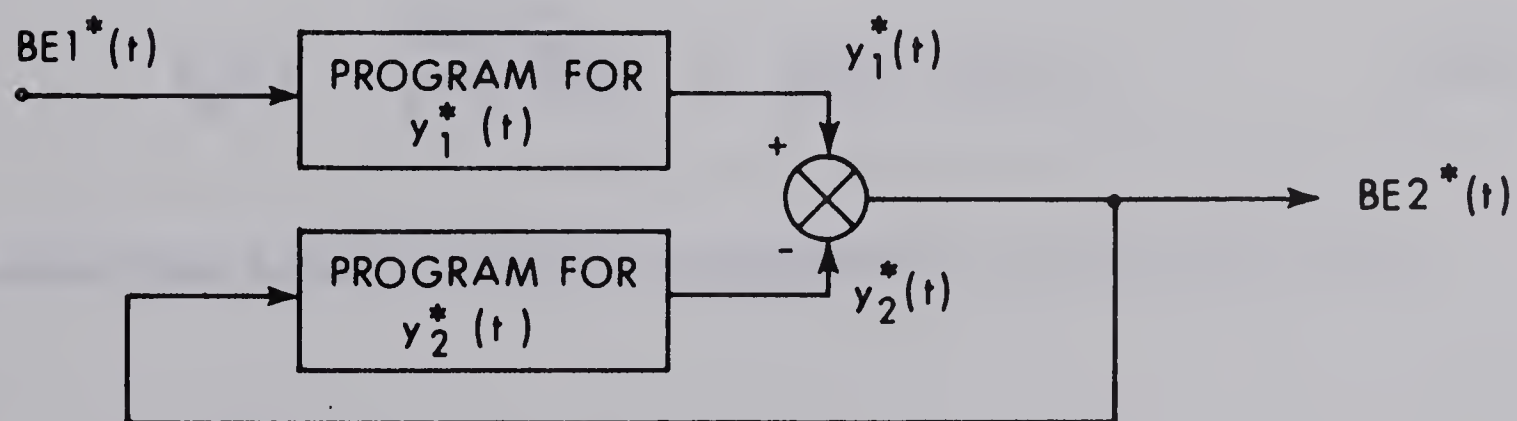
Equation (1.50) may be realized by direct digital programming.⁴ This realization is shown in FIG. 1.3. This method of realization will be used in all cases considered. It is used as it lends itself easily to programming on a hybrid or digital computer.



(a)



(b)



(c)

FIGURE 1.3 DIRECT DIGITAL PROGRAMMING REALIZATION OF $D(z)$

1-5 STATE EQUATIONS AND FLOW GRAPHS FOR

$$\underline{G_1(s) = 1/s (s + \alpha)}$$

In order to carry out some of the derivations and system simulations outlined later, it is necessary to derive the discrete state equations for the given systems.

For the plant $G_1(s)$, the entire derivation is presented. The signal flow graphs and discrete state equations of the remaining systems will then be presented without derivation.

Consider the system where $G_1(s) = 1/s (s + \alpha)$ is the plant. The signal flow graph which represents FIG. 1.1 is shown in FIG. 1.4.⁴ Applying Mason's Rule to FIG. 1.4 yields the simplified form:

$$X_1(s) = \frac{BE2(t_0^+)}{s^2 (s + \alpha)} + \frac{1}{s} X_2(t_0) + \frac{1}{s (s + \alpha)} X_2(t_0) \quad (1.51)$$

$$X_2(s) = \frac{BE2(t_0^+)}{s (s + \alpha)} + \frac{1}{s + \alpha} X_2(t_0) \quad (1.52)$$

Taking the inverse Laplace Transform of (1.51) and (1.52)

gives:

$$x_1(t) = x_1(t_0) + \frac{1}{\alpha} (1 - e^{-\alpha(t - t_0)}) x_2(t_0) + \frac{1}{\alpha} ((t - t_0) - \frac{1}{\alpha} (1 - e^{-\alpha(t - t_0)})) \text{BE2}(t_0^+)$$

$$x_2(t) = (e^{-\alpha(t - t_0)}) x_2(t_0) + \frac{1}{\alpha} (1 - e^{-\alpha(t - t_0)}) \text{BE2}(t_0^+)$$

If $t = (k + 1) T$ and $t_0 = kT$, then the previous two equations may be rewritten as:

$$x_1[(k + 1)T] = x_1[kT] + \frac{1}{\alpha} (1 - e^{-\alpha T}) x_2[kT] + \frac{1}{\alpha} (T - \frac{1}{\alpha} (1 - e^{-\alpha T})) \text{BE2}[kT] \quad (1.53)$$

$$x_2[(k + 1)T] = (e^{-\alpha T}) x_2[kT] + \frac{1}{\alpha} (1 - e^{-\alpha T}) \text{BE2}[kT] \quad (1.54)$$

Using (1.10), the following constants may be defined for (1.53) and (1.54):

$$B = (1 - A) / \alpha \quad (1.55)$$

$$C = (T - B) / \alpha \quad (1.56)$$

Substituting (1.10), (1.55) and (1.56) into (1.53) and (1.54) yields the final form of the discrete state equations for the system when $G_1(s)$ is the plant.

$$X_1[(k+1)T] = X_1[kT] + (B) X_2[kT] + (C) BE2[kT] \quad (1.57)$$

$$X_2[(k+1)T] = (A) X_2[kT] + (B) BE2[kT] \quad (1.58)$$

1-6 STATE EQUATIONS AND FLOW GRAPHS FOR $G_2(s) = 1/s (s + \alpha) (s + \beta)$

Applying the method of Section 1-5 to FIG. 1.5 yields the discrete state equations of the system containing $G_2(s)$ as follows:

$$X_1[(k+1)T] = X_1[kT] + (CONS2) X_2[kT] + (CONS3) X_3[kT] + (CONS1) BE2[kT] \quad (1.59)$$

$$X_2[(k+1)T] = (A) X_2[kT] + (CONS7) X_3[kT] + (CONS3) BE2[kT] \quad (1.60)$$

$$X_3[(k+1)T] = (B) X_3[kT] + (CONS9) BE2[kT] \quad (1.61)$$

Using (1.16), (1.17) and (1.18), the constants in the

previous three equations are defined as:

$$\gamma = \alpha + \beta \quad (1.62)$$

$$\text{CONS1} = R1 ((T - (R1)\gamma) + R1(\alpha^2 B - \beta^2 A)/(\alpha - \beta)) \quad (1.63)$$

$$\text{CONS2} = (1 - A) / \alpha \quad (1.64)$$

$$\text{CONS3} = R1 (1 + (\beta A - \alpha B) / (\alpha - \beta)) \quad (1.65)$$

$$\text{CONS7} = (A - B) / (\alpha - \beta) \quad (1.66)$$

$$\text{CONS9} = (1 - B) / \beta \quad (1.67)$$

$$\underline{\underline{1-7 \text{ STATE EQUATIONS AND FLOW GRAPHS FOR } G_3(s) = (s + \beta) / s (s + \alpha)}}$$

From FIG. 1.6, the discrete state equations of the system incorporating $G_3(s)$ are:

$$\begin{aligned} x_1[(k + 1)T] &= x_1[kT] + (\text{CONS1}) x_2[kT] + \\ &\quad (\text{CONS2}) \text{ BE2}[kT] \end{aligned} \quad (1.68)$$

$$\begin{aligned} x_2[(k + 1)T] &= (A) x_2[kT] + (\text{CONS1}) \\ &\quad \text{BE2}[kT] \end{aligned} \quad (1.69)$$

The output equation of the system for this case is:

$$C[(k+1)T] = (\beta) X_1[(k+1)T] + X_2[(k+1)T] \quad (1.70)$$

Using (1.26), the constants of (1.68) and (1.69) are defined as:

$$\text{CONS1} = (1 - A) / \alpha \quad (1.71)$$

$$\text{CONS2} = (T - \text{CONS1}) / \alpha \quad (1.72)$$

1-8 STATE EQUATIONS AND FLOW GRAPHS FOR

$$\underline{G_4(s) = 1/s (s^2 + 2\zeta \omega_n s + \omega_n^2)}$$

The discrete state equations of the system in FIG. 1.7, which contains $G_4(s)$ are:

$$\begin{aligned} X_1[(k+1)T] = & X_1[kT] + (\text{CONS1}) X_2[kT] + (\text{CONSC}) \\ & X_3[kT] + (\text{CONS2}) \text{BE2}[kT] \end{aligned} \quad (1.73)$$

$$\begin{aligned} X_2[(k+1)T] = & (\text{CONS3}) X_2[kT] + (\text{CONSB}) X_3[kT] + \\ & (\text{CONSC}) \text{BE2}[kT] \end{aligned} \quad (1.74)$$

$$\begin{aligned} X_3[(k+1)T] = & (\text{CONS4}) X_2[kT] + (\text{CONS5}) X_3[kT] + \\ & (\text{CONSB}) \text{BE2}[kT] \end{aligned} \quad (1.75)$$

Using the constants defined in (1.30), (1.31), (1.32),

(1.37) and (1.38), the constants in equations (1.73), (1.74) and (1.75) are defined as:

$$\phi = \tan^{-1} (-A/B) \quad (1.76)$$

$$\text{CONSA} = (\text{AET}) \cos(BT) \quad (1.77)$$

$$\text{CONSB} = ((\text{AET}) \sin(BT)) / B \quad (1.78)$$

$$\text{CONSC} = \text{AC1} (1 - ((\text{AET}) \cos(BT + \phi)) / \cos(\phi)) \quad (1.79)$$

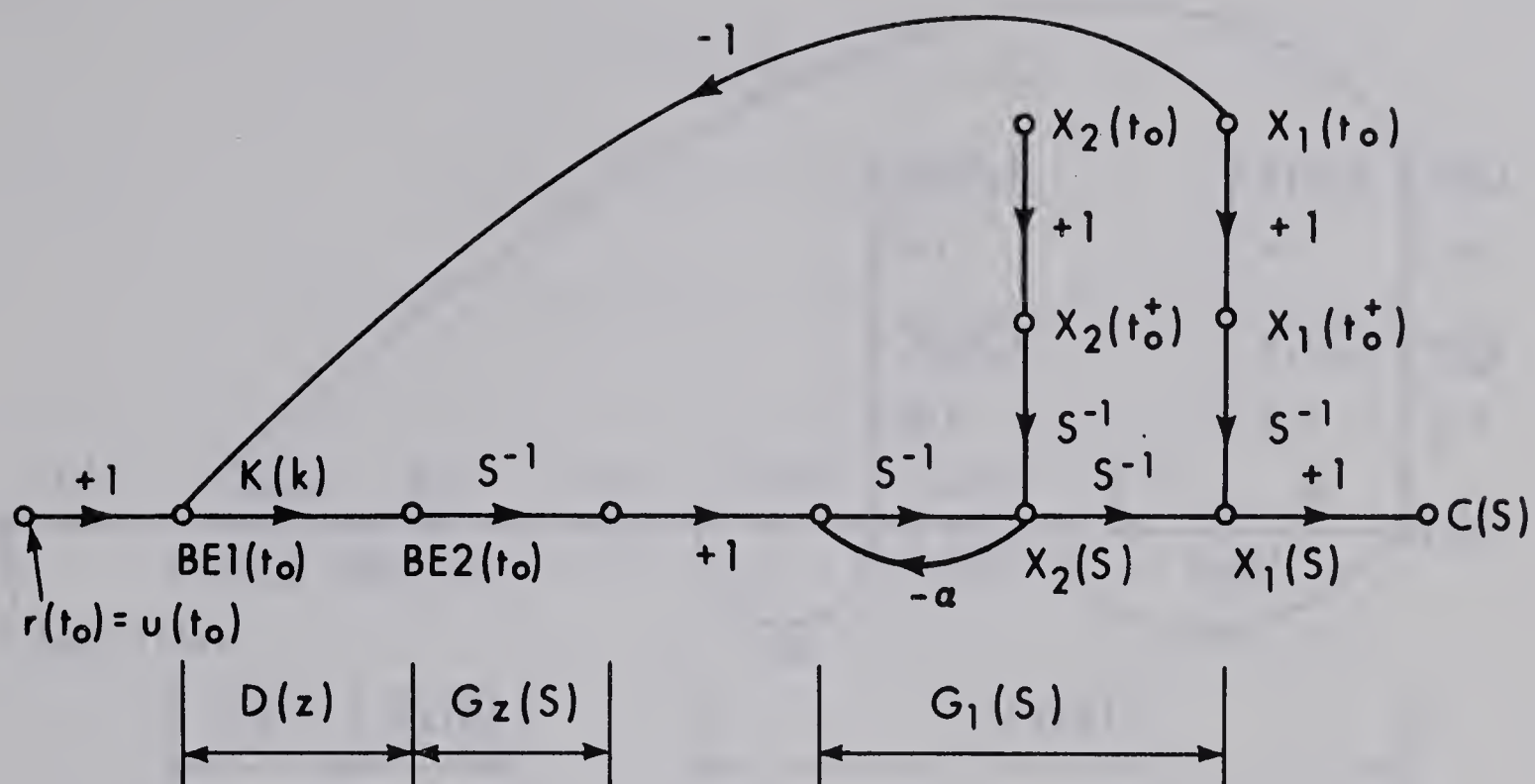
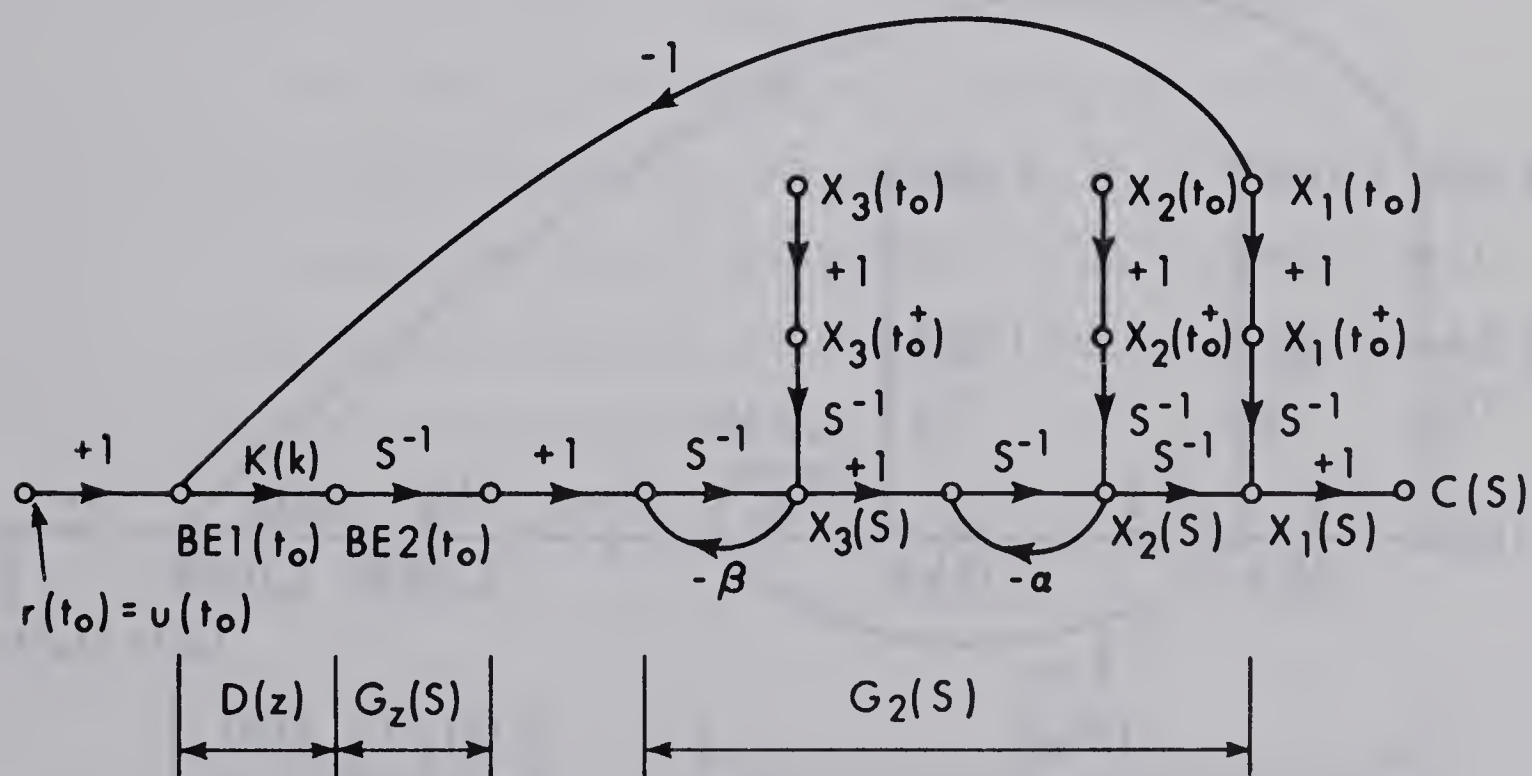
$$\text{CONS1} = \text{CONSB} + 2(A) \text{CONSC} \quad (1.80)$$

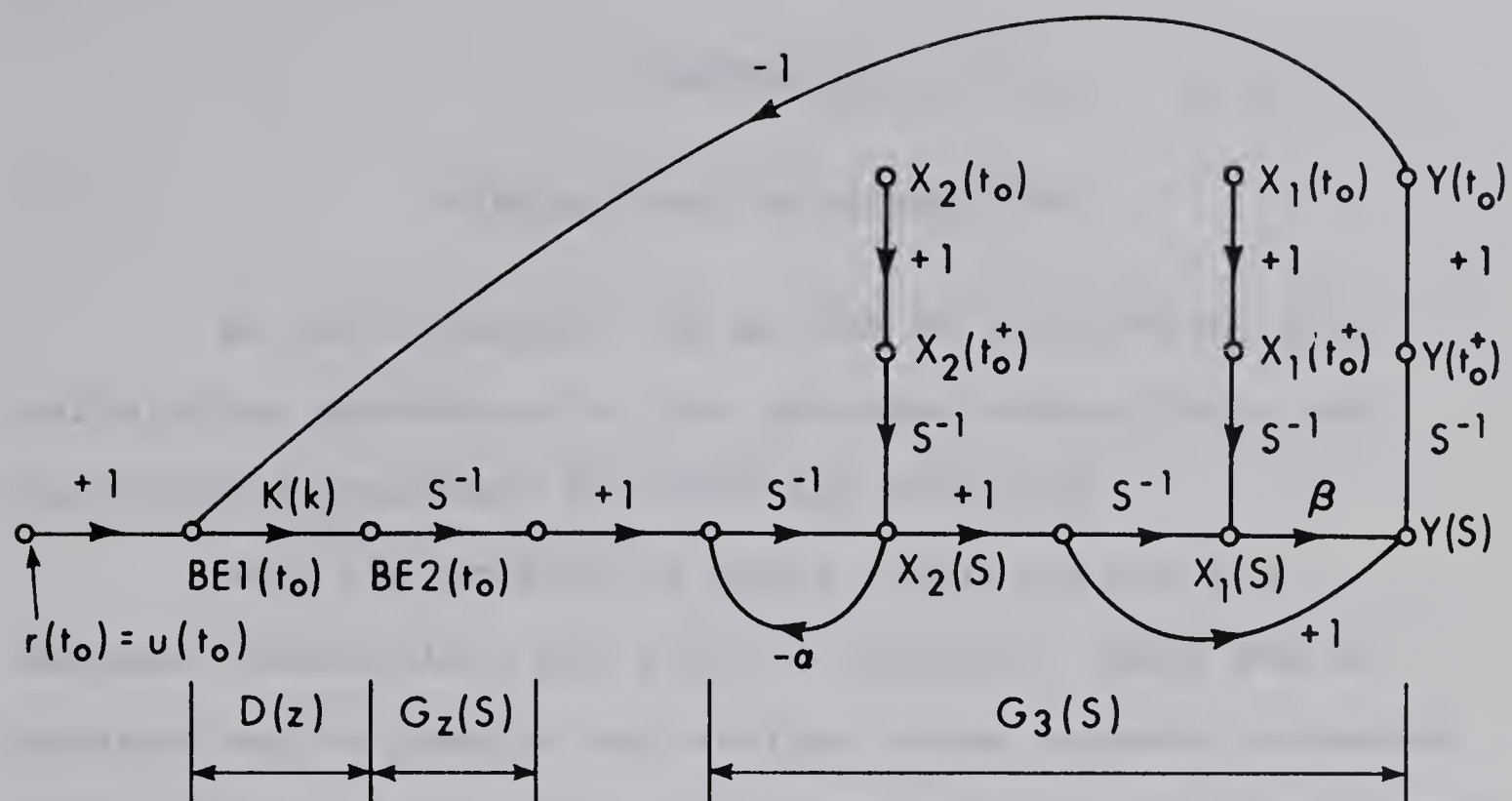
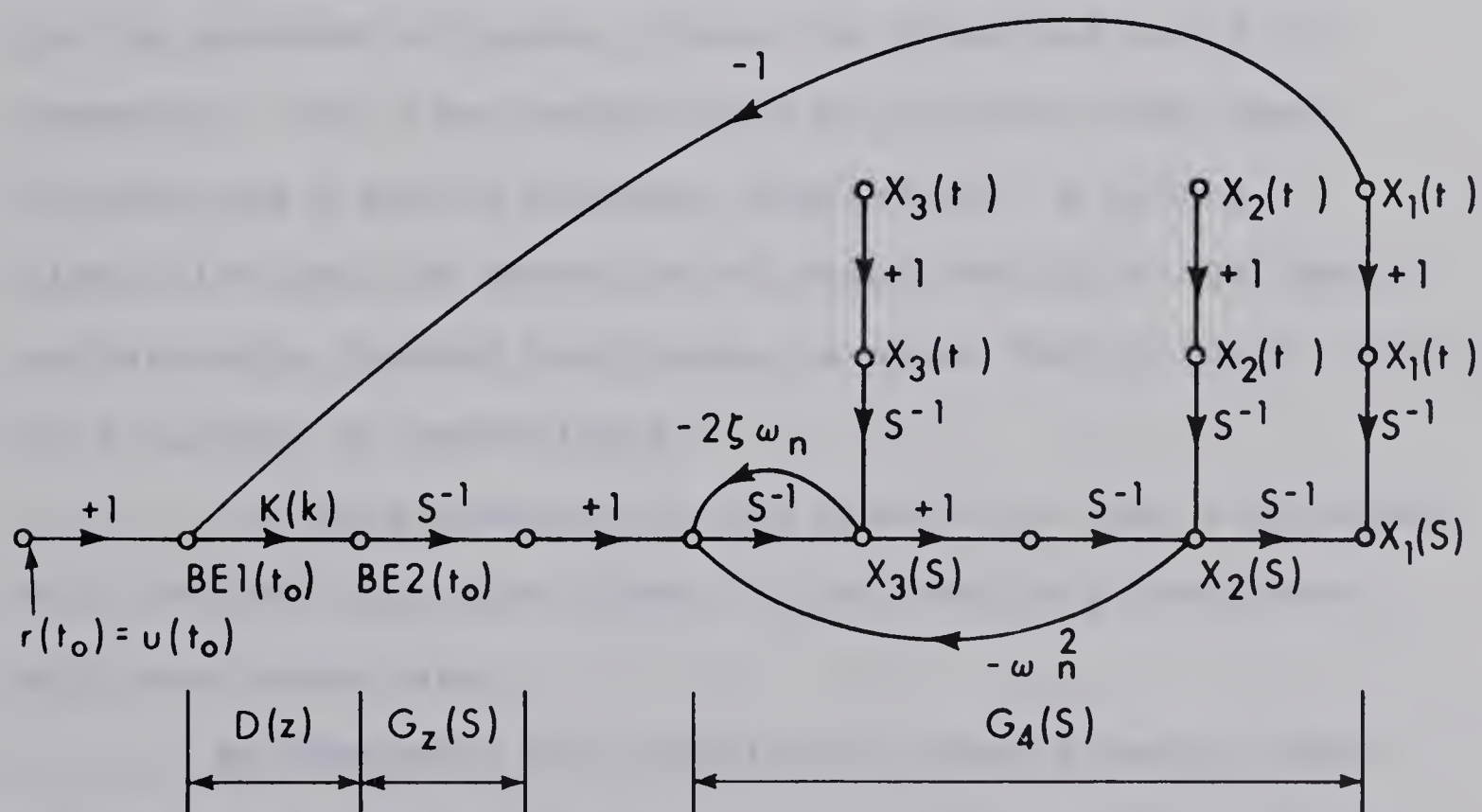
$$\begin{aligned} \text{CONS2} = \text{AC1} (T + 2(A) \text{AC1} (\text{CONSA} - 1) + \\ \text{CONSB} (2(A^2) \text{AC1} - 1)) \end{aligned} \quad (1.81)$$

$$\text{CONS3} = \text{CONSA} + (A) \text{CONSB} \quad (1.82)$$

$$\text{CONS4} = -\omega_n^2 \text{CONSB} \quad (1.83)$$

$$\text{CONS5} = \text{CONSA} - (A) \text{CONSB} \quad (1.84)$$

FIGURE 1.4 FLOW GRAPH FOR SYSTEM CONTAINING $G_1(s)$ FIGURE 1.5 FLOW GRAPH FOR SYSTEM CONTAINING $G_2(s)$

FIGURE 1.6 FLOW GRAPH FOR SYSTEM CONTAINING $G_3(s)$ FIGURE 1.7 FLOW GRAPH FOR SYSTEM CONTAINING $G_4(s)$

CHAPTER II

HYBRID COMPUTER SIMULATION

In this chapter, the method of simulation, the calculation procedure for the extended controllers, and the testing procedure followed are detailed.

When the problem of noise reduction for the deadbeat controllers was first considered, there was no obvious way to predict the minimum noise transfer extended controllers. It was also not known whether any significant noise transfer reduction was obtainable by this extension method.

By observing sequential sets of extended controllers in the presence of noise, these two questions could be answered. The first method used to perform these observations was a hybrid computer simulation. A hybrid simulation had the advantage of using available equipment and straight forward techniques to allow fairly rapid testing of a variety of controllers.

For this simulation, the plant $G_1(s)$ was considered with one and two extra terms. $G_2(s)$ was only considered with one extra term.

By observing the results for these plants, information which would lead to a criterion for predicting the minimum noise transfer extended controllers was obtained.

2-1 HYBRID COMPUTER SYSTEM

The computers used in the experimental simulation were:

1. A Digital Equipment Corporation PDP-8 programmed in FOCAL.
2. An Electronic Associates Incorporated TR-48 analog computer.
3. A Redcor Corporation Series 610 Linkage System interface.

The PDP-8 digital computer is a 4K core memory version with external magnetic tape drives.

FOCAL, which is an interpretive language, was chosen as the programming language because of the ease of programming. Although FOCAL is much slower than machine language, it was felt that the ease with which the new values for various controllers could be inserted in the program compensated for the slowness. The maximum sampling rate the program could operate at was $T = 0.5$ second. This was not a disadvantage as the normal sampling rate used was 1 second.

The TR-48 was used since it has external mode control on each integrator pair. This allows the PDP-8 to control the operating state of any part of the simulated system.

2-2 HYBRID SYSTEM SIMULATION

In order to implement the problem on the hybrid computer, it was first necessary to write a digital computer program to simulate the $D(z)$. This digital program was also written to include:

1. Control of the start and duration of the experimental runs.
2. Control of the operating state of the TR-48.
3. Control of the actual sampling rate T .
4. The zero-order hold function which is shown separately in FIG. 1.1 as $G_z(s)$.
5. The calculation of the root-mean-square error for the system upon completion of the experimental run.

The derivation and listing of this program is given in Appendix A.

The measuring criterion used is the root-mean-square error of the system. Therefore it is necessary to measure the difference between the response to a unit step with and without the noise input. For a particular controller under test, this may be accomplished as shown in FIG. 2.1. In FIG. 2.1, $c_m(t)$ is the desired system response to a unit step as shown in FIG. 1.2.

The test system shown in FIG. 2.1 is difficult to implement as it requires duplication of the system to yield

$c'(t)$. It is also somewhat redundant as the value of $c_m(t)$ is unity for all but the first n sampling periods. Therefore it is possible to alter the hybrid realization to that shown in FIG. 2.2. This yields the same results as the system in FIG. 2.1.

The root-mean-square error of the system is

$\left[\overline{[c'(t)]^2} \right]^{\frac{1}{2}}$. This assumes that the noise source is the same for any controller measured. Since the noise source may fluctuate slightly, the root-mean-square error is redefined for this test. Using the new definition, the root-mean-square error is actually the ratio of the root-mean-squares of the output of the system and the noise input. This allows direct comparison of the results obtained for any controller.

Redefining the root-mean-square error yields:

ROOT-MEAN-SQUARE ERROR \equiv RMSE =

$$\left[\overline{[c'(t)]^2} / \overline{[n(t)]^2} \right]^{\frac{1}{2}} \quad (2.1)$$

In (2.1), $\overline{[c'(t)]^2}$ and $\overline{[n(t)]^2}$ are defined as:

$$\overline{[c'(t)]^2} = \frac{1}{T_T} \int_0^{T_T} [c'(t)]^2 dt \quad (2.2)$$

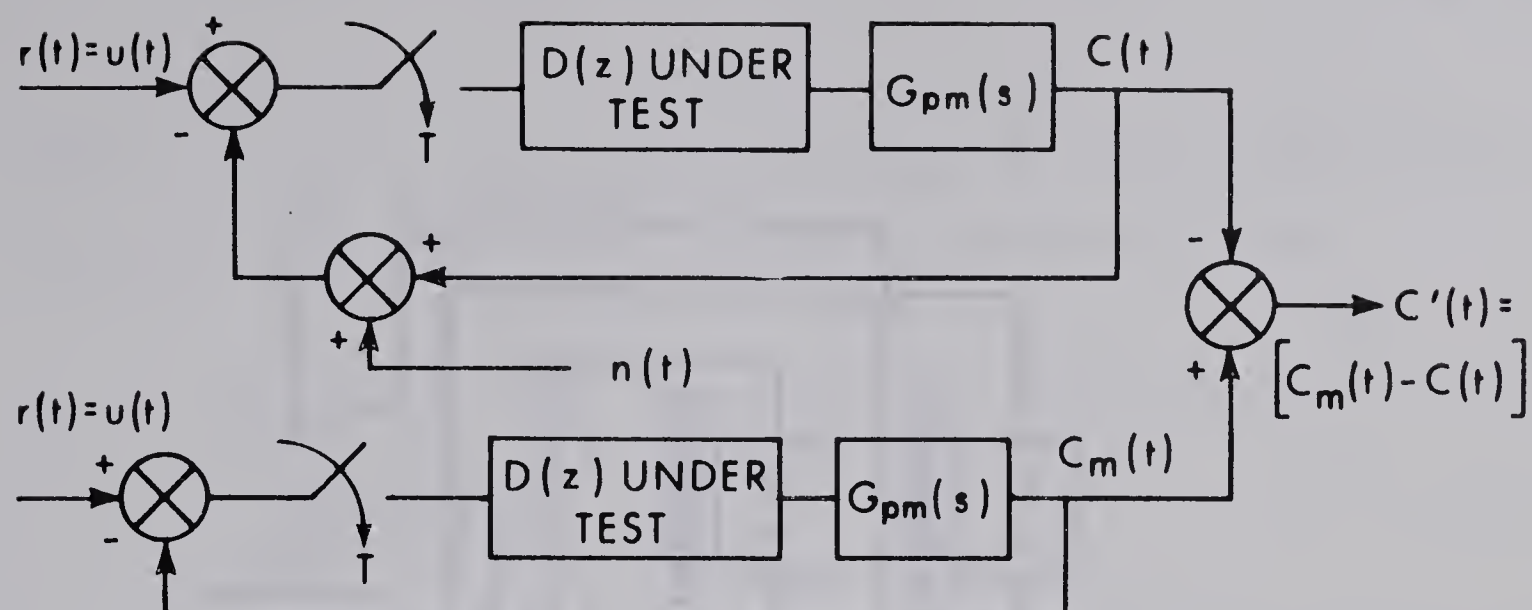


FIGURE 2.1 TEST SYSTEM TO MEASURE SYSTEM ERROR WHEN NOISE PRESENT

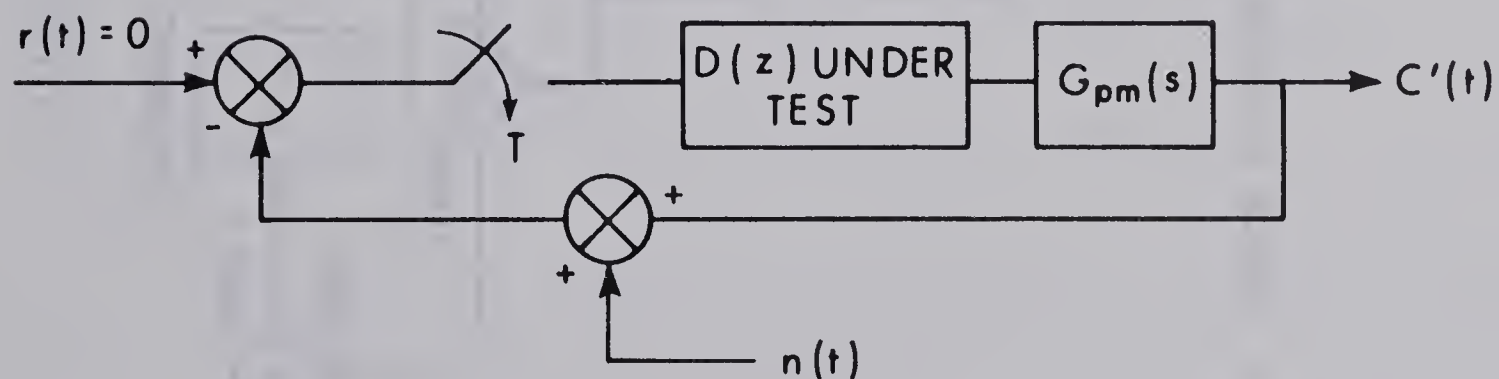


FIGURE 2.2 TEST SYSTEM IMPLEMENTED ON HYBRID TO MEASURE SYSTEM ERROR WHEN NOISE PRESENT

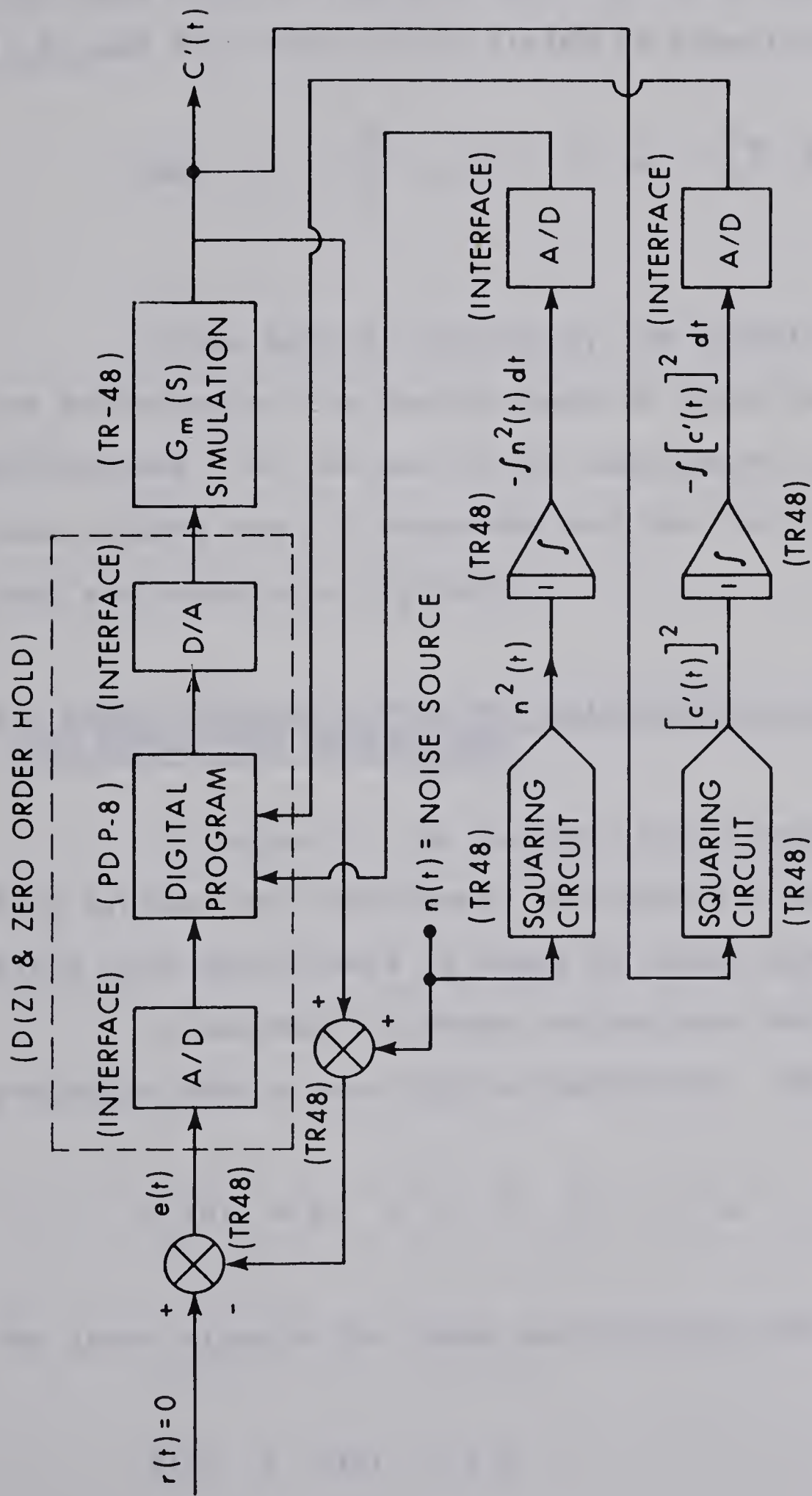


FIGURE 2.3 BLOCK DIAGRAM OF HYBRID COMPUTER SIMULATION

$$\overline{n^2(t)} = \frac{1}{T_T} \int_0^{T_T} n^2(t) dt \quad (2.3)$$

where T_T = total time of the experimental run. Substituting (2.2) and (2.3) into (2.1) yields in simplified form:

$$\text{RMSE} = \left[\int_0^{T_T} [c'(t)]^2 dt / \int_0^{T_T} n^2(t) dt \right]^{\frac{1}{2}} \quad (2.4)$$

In the hybrid simulation, the squaring and integration are performed on the analog computer using the electronic multipliers. At the end of the experimental run, the two mean-squares are A/D converted and the division and square root are calculated digitally.

2-3 STATE VARIABLE METHOD FOR CALCULATING MINIMUM AND EXTRA TERM CONTROLLERS

In Chapter 1, the discrete state equations for the four systems were developed. A method for calculating the extra term controllers is based on these equations.

It assumed for these derivations that all state variables have a zero initial condition. That is:

$$x_i(0) = 0; \quad i = 1, 2, \dots, m \quad (2.5)$$

The input signals for these calculations are:

$$r(t) = u(t) = 1.0 \quad (2.6)$$

$$n(t) = 0 \quad (2.7)$$

If the system is to reach deadbeat in n sampling periods, then the following values of the state variables must be reached.

$$X_1[(n+1)T] = 1.0 \quad (2.8)$$

$$X_i[(n+1)T] = 0; i = 2, 3, \dots, m \quad (2.9)$$

The method of calculation is derived for the plant $G_1(s) = 1/s(s + \alpha)$.

From (1.57) and (1.58), the discrete state equations for the system containing $G_1(s)$ are:

$$X_1[(k+1)T] = X_1[kT] + (B) X_2[kT] + (C) \text{BE2}[kT]$$

$$X_2[(k+1)T] = (A) X_2[kT] + (B) \text{BE2}[kT]$$

From FIG. 1.5, the quantity $\text{BE1}(t_0^+)$ may be defined as:

$$\text{BE1}(t_0^+) = r(t_0) - X_1(t_0) = 1 - X_1(t_0) \quad (2.10)$$

As in section 1-5, define $t_0 = kT$ to allow (2.10) to be expressed in discrete form. Thus (2.10) becomes:

$$\text{BE1}[kT] = 1 - X_1[kT] \quad (2.11)$$

The digital controller may be replaced in a discrete state equation by the equivalent gain during any sampling period.⁴ This equivalent gain is defined for the k^{th} sampling period as:

$$K(k) = h_k / e_k \quad (2.12)$$

where h_k and e_k are the controller coefficients as shown in (1.1). This equivalent gain is shown in FIG. 1.4 to FIG. 1.7. From these diagrams it may be seen that:

$$BE2[kT] = K(k) (BE1[kT]) \quad (2.13a)$$

Substituting (2.11) into (2.13) yields the controller output in terms of $K(k)$ as:

$$BE2[kT] = K(k) (1 - X_1[kT]) \quad (2.13b)$$

Substituting (2.13) into (1.57) and (1.58) yields the discrete state equations as:

$$\begin{aligned} X_1[(k+1)T] &= (1 - (A) K(k)) X_1[kT] + (B) \\ &\quad X_2[kT] + (A) K(k) \end{aligned} \quad (2.14)$$

$$\begin{aligned} X_2[(k+1)T] &= -K(k) (B) X_1[kT] + (A) X_2[kT] + \\ &\quad (B) K(k) \end{aligned} \quad (2.15)$$

The minimum time controller for $G_1(s)$ reaches

deadbeat at $k = 1$. This controller is unique and may be calculated from (2.14) and (2.15) by use of the initial conditions in (2.5), (2.8) and (2.9). Following is the calculation of this controller.

The state variables at the end of the sampling period where $k = 0$ are:

$$x_1[T] = (A) K(0) \quad (2.16)$$

$$x_2[T] = (B) K(0) \quad (2.17)$$

Similarly for $k = 1$, the state variables are:

$$\begin{aligned} x_1[2T] &= (1 - (A) K(1)) x_1[T] + (B) \\ &\quad x_2[T] + (A) K(1) \end{aligned} \quad (2.18)$$

$$\begin{aligned} x_2[2T] &= -K(1) (B) x_1[T] + (A) \\ &\quad x_2[T] + (B) K(1) \end{aligned} \quad (2.19)$$

From (2.8) and (2.9), the final values are:

$$\begin{aligned} x_1[2T] &= 1.0 \\ x_2[2T] &= 0 \end{aligned} \quad (2.20)$$

Substituting (2.16), (2.17) and (2.20) into (2.18) and (2.19) yields the simplified state equations:

$$x_1[2T] = 1.0 = (A + B^2) K(0) + (A) K(1) + (-A^2) K(0) K(1) \quad (2.21)$$

$$x_2[2T] = 0 = (AB) K(0) + (B) K(1) + (-AB) K(0) K(1) \quad (2.22)$$

Solving (2.21) and (2.22) yields:

$$K(0) = 1 / (A (1 - A) + B^2) \quad (2.23)$$

$$K(1) = A / (A^2 - B^2) \quad (2.24)$$

The expression (2.11) defines the error $e(t)$ shown in FIG. 1.1 when $n(t) = 0$. From FIG. 1.1, this error in general form is:

$$e[T] = BE_1[T] = 1 - x_1[T] \quad (2.25)$$

It can be shown that the denominator terms of a deadbeat digital controller are the same as the system error.

Therefore e_1 of the controller may be written as:

$$e_1 = e[T] = 1 - x_1[T] \quad (2.26)$$

Substituting (2.16) and (2.23) into (2.26) and solving yields:

$$e_1 = A / (A (1 - A) + B^2) \quad (2.27)$$

The coefficients of the minimum time deadbeat digital controller for $G_1(s) = 1/s (s + \alpha)$ may be calculated from (2.12), (2.23), (2.24) and (2.27) to yield in simplified form:

$$h_0 = K(0) = 1 / (A (1 - A) + B^2) \quad (2.28)$$

$$h_1 = e_1 K(1) = \frac{A^2}{(A (1 - A) + B^2)} \quad (2.29)$$

$$e_1 = A / (A (1 - A) + B^2) \quad (2.30)$$

where A and B are defined in (1.16) and (1.17).

Since the minimum time deadbeat controller is unique, no noise reduction can be performed on it without altering its form. The method considered involves adding extra terms to the controller without losing the other deadbeat characteristics. In this way, additional degrees of freedom in the design of the controller are available. These may possibly be used to improve the noise performance of the system.

Because additional degrees of freedom are available in the controller design, an infinite number of possible controllers satisfy the extended deadbeat requirements. Some method of predicting which controllers, if any, have the best noise performance must be derived. This is where the hybrid simulation is useful. The performance of various

extended controllers may be appraised in the presence of noise.

The deadbeat controller which is extended by one term has the following form for the system containing $G_1(s)$:

$$D(z) = \frac{h_0 + h_1 z^{-1} + h_2 z^{-2}}{1 + e_1 z^{-1} + e_2 z^{-2}} \quad (2.31)$$

In (2.31), e_2 and h_2 are the extra controller terms. The following derivation will show that it is possible to express h_0 , h_1 , h_2 and e_2 as functions of e_1 , α and T . By extending the controller by one extra term, one extra degree of freedom is available as e_1 becomes the only unknown.

For a deadbeat controller, e_1 is the system error at the end of the first sampling period. Because of the deadbeat constraint, e_1 must always lie between 0 and 1.0. All of the controllers represented by (2.31) may thus be solved by inserting all values of e_1 into (2.28) to (2.30). This is of course not practical.

Instead a small number of controllers with various consecutive values of e_1 will be tested on the hybrid. Therefore extended controllers of the form (2.31) will be solved for by "iterating" e_1 over the range $0 \leq e_1 \leq 1.0$ for given values of α and T .

As for the minimum time deadbeat controller, the discrete state equations are written for the various

sampling periods. For $k = 0$, the state variables are:

$$X_1[T] = (A) K(0) \quad (2.32)$$

$$X_2[T] = (B) K(0) \quad (2.33)$$

For $k = 1$, the state variables are:

$$\begin{aligned} X_1[2T] &= (1 - (A) K(1)) X_1[T] + (B) \\ &\quad X_2[T] + (A) K(1) \end{aligned} \quad (2.34)$$

$$\begin{aligned} X_2[2T] &= -K(1) (B) X_1[T] + (A) \\ &\quad X_2[T] + (B) K(1) \end{aligned} \quad (2.35)$$

For $k = 2$, the state variables are:

$$\begin{aligned} X_1[3T] &= (1 - (A) K(2)) X_1[2T] + (B) \\ &\quad X_2[2T] + (A) K(2) \end{aligned} \quad (2.36)$$

$$\begin{aligned} X_2[3T] &= -K(2) (B) X_1[2T] + (A) \\ &\quad X_2[2T] + (B) K(2) \end{aligned} \quad (2.37)$$

The final values of the state variables are:

$$\begin{aligned} X_1[3T] &= 1.0 \\ X_2[3T] &= 0 \end{aligned} \quad (2.38)$$

Substituting (2.32) to (2.35) and (2.38) into (2.36) and (2.37) yields in simplified form:

$$\begin{aligned}
 X_1[3T] &= 1.0 = (A + B^2 + AB^2) K(0) + \\
 &\quad (A + B^2) K(1) + (A) K(2) + (-A^2 - AB^2) \\
 &\quad K(0) K(1) + (-A^2 - AB^2) K(0) K(2) + \\
 &\quad (-A^2) K(1) K(2) + (A^3) K(0) K(1) K(2)
 \end{aligned} \tag{2.39}$$

$$\begin{aligned}
 X_2[3T] &= 0 = (A^2 B) K(0) + (AB) K(1) + (B) \\
 &\quad K(2) + (-A^2 B) K(0) K(1) + (-AB - B^3) \\
 &\quad K(0) K(2) + (-AB) K(1) K(2) + \\
 &\quad (A^2 B) K(0) K(1) K(2)
 \end{aligned} \tag{2.40}$$

Eliminate $K(2)$ and all its products from the above expressions in the following manner:³

1. MULTIPLY (2.39) by B
2. MULTIPLY (2.40) by $-A$
3. Adding the results of (1) and (2) yields the

simplified expression:

$$\begin{aligned}
 B &= K(0) (A + B^2 + AB^2 - A^3) (B) + K(1) (A + B^2 - A^2) \\
 &\quad (B) + K(0) K(1) (A^2 - AB^2 + A^3) (B)
 \end{aligned} \tag{2.41}$$

Solving for $K(1)$ in terms of $K(0)$ from (2.41) yields:

$$K(1) = \frac{(1 - K(0) (A + B^2 + AB^2 - A^3))}{((A + B^2 - A^2) + K(0) (A^3 - AB^2 - A^2))} \quad (2.42)$$

$K(2)$ may be expressed from (2.40) in terms of $K(0)$ and $K(1)$ in the following form:

$$K(2) = \frac{(K(0) (-A^2B) + K(1) (-AB) + K(0) K(1) (A^2B))}{(B + K(0) (-AB - B^3) + K(1) (-AB) + K(0) K(1) (A^2B))} \quad (2.43)$$

As stated in (2.26):

$$e_k = 1 - X_1[kT] \quad (2.44)$$

Substituting for k in (2.44) yields the following expressions:

$$e_1 = 1 - X_1[T] \quad (2.45)$$

$$e_2 = 1 - X_1[2T] \quad (2.46)$$

Substituting (2.32) into (2.45) yields $K(0)$ as the following function of e_1 :

$$K(0) = (1 - e_1) / A \quad (2.47)$$

If e_1 is iterated over the range 0 to 1, all possible values of $K(0)$ are obtained. These in turn allow the calculation of all possible values of $K(1)$ and $K(2)$ from (2.42) and (2.43). Substituting (2.34) into (2.46) yields e_2 as a function of $K(0)$ to $K(2)$ which are now functions of e_1 . The coefficient e_2 is obtained as:

$$e_2 = 1 + K(0) (-A - B^2) + K(1) (-A) + K(0) K(1) (A^2) \quad (2.48)$$

Table 2.1(a) lists the coefficients of the minimum time deadbeat for $G_1(s)$. The extended controller for $G_1(s)$ is given in Table 2.1(b). These coefficients in Table 2.1(b) can all be expressed as functions of e_1 .

\underline{k}	$\underline{h_k} = \underline{K(k) (e_k)}$	$\underline{e_k}$
0	$h_0 = \frac{1}{(A(1-A) + B^2)}$	1
1	$h_1 = \frac{A^2}{(A(1-A) + B^2)(A^2 - B^2)}$	$e_1 = \frac{A}{(A(1-A) + B^2)}$

TABLE 2.1(a) COEFFICIENTS OF MINIMUM DEADBEAT
CONTROLLER FOR $G_1(s) = 1/s (s + \alpha)$

\underline{k}	$\underline{h_k}$	$\underline{e_k}$
0	$h_0 = \frac{(1 - e_1)}{A}$	1.0
1	$h_1 = e_1 \left[\frac{1 - K(0)(A+B)^2 + AB^2 - A^3}{(A+B^2 - A^2) + K(0)(A^3 - AB^2 - A^2)} \right]$	$0 \leq e_1 \leq 1 \text{ (ITERATED)}$
2	$h_2 = e_2 \left[\frac{K(0)(-A^2B) + K(1)(-AB) + K(0)K(1)(A^2B)}{(B + K(0)(-AB - B^3) + K(1)(-AB) + K(0)K(1)(A^2B))} \right]$	$e_2 = 1 + K(0)(-A - B^2) + K(1)(-A) + K(0)K(1)(A^2)$

TABLE 2.1(b) COEFFICIENTS OF 3 TERM DEADBEAT CONTROLLER FOR $G_1(s) = 1/s(s + \alpha)$ (ONE EXTRA TERM)

The resultant equations listed in Table 2.1(b) were programmed in Fortran and the appropriate iteration in e_1 carried out. This calculation yielded a set of controllers, with one extra term, for simulation on the hybrid system.

2-4 z-DOMAIN METHOD FOR CALCULATING MINIMUM AND EXTRA TERM CONTROLLERS

The state variable method, outlined in Section 2-3, is too cumbersome for higher order systems and thus another method was used. This method is based on the z-domain system transfer function. For a unity feedback system, the system transfer function is of the form:

$$\frac{C(z)}{R(z)} = \frac{D(z) G_{pm}(z)}{1 + D(z) G_{pm}(z)} \quad (2.49)$$

It can be shown for a system with deadbeat response to a unit step, that the conditions on the state variables given in (2.8) and (2.9) are met if the following equation is satisfied:

$$1 + D(z) G_{pm}(z) = z^n \quad (2.50)$$

where n is the number of sampling periods required to reach deadbeat.⁵

To illustrate the use of this method, consider the example of $G_2(s) = 1/s (s + \alpha) (s + \beta)$. From (1.15), the

z - transform of $G_{p2}(s)$ is:

$$G_{p2}(z) = \frac{(R7) z^2 + (R8) z + U}{(z - 1)(z - A)(z - B)} \quad (1.15)$$

Assume the minimum deadbeat controller is the following form:

$$D(z) = \frac{h_0 (z - A)(z - B)}{z^2 + e_1 z + e_2} = \frac{h_0 - h_0(A + B) z^{-1} + h_0(AB) z^{-2}}{1 + e_1 z^{-1} + e_2 z^{-2}} \quad (2.51)$$

Therefore the numerator coefficients of the controller are defined from (2.51) as:

$$\begin{aligned} h_0 &= h_0 \\ h_1 &= -h_0(A + B) \\ h_2 &= h_0(AB) \end{aligned} \quad (2.52)$$

The digital controller is assumed to be the form (2.51), because for deadbeat to occur the $D(z)$ must cancel all the poles of $G_{pm}(z)$ except the pole at $z = -1$.⁵

For $G_2(s)$, the minimum deadbeat occurs at $n = 3$. Substituting (1.15) and (2.51) into (2.50) yields the

simplified form:

$$\begin{aligned}
 & z^3 [1 + h_0 (R2 + R3 + R4)] + z^2 [h_0 (R1)T - h_0 (R2) \\
 & (1 + A + B) - h_0 (R3) (2 + B) - h_0 (R4) \\
 & (2 + A) + (e_1 - 1)] + z [- h_0 (R1)T (A + B) + \\
 & h_0 (R2) (A + B + AB) + h_0 (R3) (1 + 2B) + \\
 & h_0 (2A + 1) R4 + (e_2 - e_1)] + [h_0 (R1) T(AB) - \\
 & h_0 (R2) AB - h_0 (R3)B - h_0 (R4) A - e_2] = \\
 & z^6 + [e_1 - 1] z^5 + [e_2 - e_1] z^4 - \\
 & [e_2] z^3
 \end{aligned} \tag{2.53}$$

Equating the coefficients of z^n in (2.53) results in the following relationships:

For $n = 3$:

$$1 + h_0 (R2 + R3 + R4) = 0 \tag{2.54}$$

For $n = 2$:

$$\begin{aligned}
 & h_0 (R1)T - h_0 (R2) (1 + A + B) - h_0 (R3) (2 + B) - \\
 & h_0 (R4) (2 + A) + (e_1 - 1) = 0
 \end{aligned} \tag{2.55}$$

For $n = 1$:

$$\begin{aligned}
 & -h_0 (R1)T (A + B) + h_0 (R2) (A + B + AB) + h_0 (R3) \\
 & (1 + 2B) + h_0 (2A + 1) R4 + \\
 & (e_2 - e_1) = 0
 \end{aligned} \tag{2.56}$$

For $n = 0$:

$$h_0 (R1)T (AB) - h_0 (R2)AB - h_0 (R3)B - h_0 (R4)A - e_2 = 0 \quad (2.57)$$

To simplify the manipulation of these equations, define the following terms:

$$R5 = (R1)T (AB) - (R2)AB - (R3)B - (R4)A \quad (2.58a)$$

$$R6 = R4 (2 + A) + R3 (2 + B) + R2 (1 + A + B) - (R1)T \quad (2.58b)$$

Solving (2.54) to (2.57) and using (2.58) to simplify the result yields the controller coefficients as:

$$h_0 = 1 / (R5 - R6 - (R1)T (A + B) + R2 (A + B + AB) + R3 (2B + 1) + R4 (1 + 2A)) \quad (2.59)$$

$$e_1 = 1 + h_0 (R6) \quad (2.60)$$

$$e_2 = h_0 (R5) \quad (2.61)$$

These results are tabulated in Table 2.2(a).

The controller with one extra term is derived in a similar manner. As in Section 1-3, the value of e_1 is iterated between 0 and 1 to solve the extra unknown.

Equation (2.50) when the extended controller is sought becomes:

$$1 + D(z) G_{p2}(z) = z^4 \quad (2.62)$$

Assume the following controller form:

$$D(z) = \frac{h_0 (z - A) (z - B) (z - C)}{z^3 + e_1 z^2 + e_2 z + e_3} = \frac{h_0 + h_1 z^{-1} + h_2 z^{-2} + h_3 z^{-3}}{1 + e_1 z^{-1} + e_2 z^{-2} + e_3 z^{-3}} \quad (2.63)$$

where C is the additional unknown zero of D(z).

Defining the numerator coefficients of the controller from (2.63) yields:

$$h_0 = h_0$$

$$h_1 = -h_0 (A + B + C)$$

$$h_2 = h_0 (AB + AC + BC)$$

$$h_3 = -h_0 ABC \quad (2.64)$$

Substituting (1.15) and (2.63) into (2.62) yields the simplified result:

$$\begin{aligned}
 & z^4 + z^3[e_1 + h_0(R7) - 1] + z^2[e_2 - e_1 + h_0(R8) - \\
 & h_0 C(R7)] + z[e_3 - e_2 + h_0(U) - \\
 & h_0 C(R8)] + [-e_3 - h_0 CU] = \\
 & z^8 + z^7[e_1 - 1] + z^6[e_2 - e_1] + \\
 & z^5[e_3 - e_2] + z^4[-e_3]
 \end{aligned} \tag{2.65}$$

Equating the coefficients of z^3 yields:

$$e_1 + h_0(R7) - 1 = 0$$

From this equation h_0 is calculated to be:

$$h_0 = (1 - e_1) / R7 \tag{2.66}$$

Since e_1 is iterated from 0 to 1, it is now only necessary to solve for the other coefficients in terms of h_0 . Equating the coefficients of z^n in (2.65) and solving yields:

$$C = 1 - 1/(h_0(R7 + R8 + U)) \tag{2.67}$$

$$e_2 = h_0(U - C(R8 + U)) \tag{2.68}$$

$$e_3 = -h_0 CU \tag{2.69}$$

\underline{k}	$\underline{h_k}$	$\underline{e_k}$
0	$h_0 = 1 / (R5 - R6 - (R1)T(A+B) + R2(A+B+AB) + R3(2B+1) + R4(2A+1))$	1.0
1	$h_1 = -h_0(A+B)$	$e_1 = 1 + h_0(R6)$
2	$h_2 = h_0AB$	$e_2 = h_0(R5)$

TABLE 2.2 (a) COEFFICIENTS OF MINIMUM DEADBEAT CONTROLLER
FOR $G_2(s) = 1/s(s + \alpha)(s + \beta)$

\underline{k}	$\underline{h_k}$	$\underline{e_k}$
0	$h_0 = (1 - e_1)/R7$	1.0
1	$h_1 = -h_0(A+B+C)$	$0 \leq e_1 \leq 1$ (ITERATED)
2	$h_2 = h_0(AB + AC + BC)$	$e_2 = h_0(U - C(R8 + U))$
3	$h_3 = -h_0(ABC)$	$e_3 = -h_0 CU$

TABLE 2.2 (b) COEFFICIENTS OF 4 TERM DEADBEAT CONTROLLER
FOR $G_2(s) = 1/s(s + \alpha)(s + \beta)$ (ONE EXTRA TERM)

These results are summarized in Table 2.2(b). All these coefficients are calculable using e_1 as the only extra unknown.

2-5 NOISE SOURCE USED IN HYBRID SIMULATION

The random noise generator used to obtain the experimental data is a SERVOMEX TYPE R.G. 77. The noise spectrum of this generator is shown in FIG. 2.4. The low frequency output of this generator is flat to d.c. according to the manufacturer's specifications.

It is known from sampling theory that any continuous input signal can be represented by the bandwidth $-f_s/2$ to $f_s/2$ after sampling, where f_s is the sampling frequency. Also the input signal must be strictly bandlimited between $-f_s/2$ and $f_s/2$ for the sampling process to uniquely reconstruct the input.⁶ If the input signal is pure white noise and is strictly bandlimited, the output of the sampler is bandlimited white noise. The auto-spectral density is shown in FIG. 2.5 for bandlimited white noise.⁷

No suitable low pass filter was available for this experiment. There was however a Krohn-Hite 330M tunable bandpass filter available. This filter has a slope of -80 db/decade and a tunable range from 0.2 Hz to 20 KHz. In order to generate an approximation to low-passed white noise, the aliasing properties of the sampling process are used.

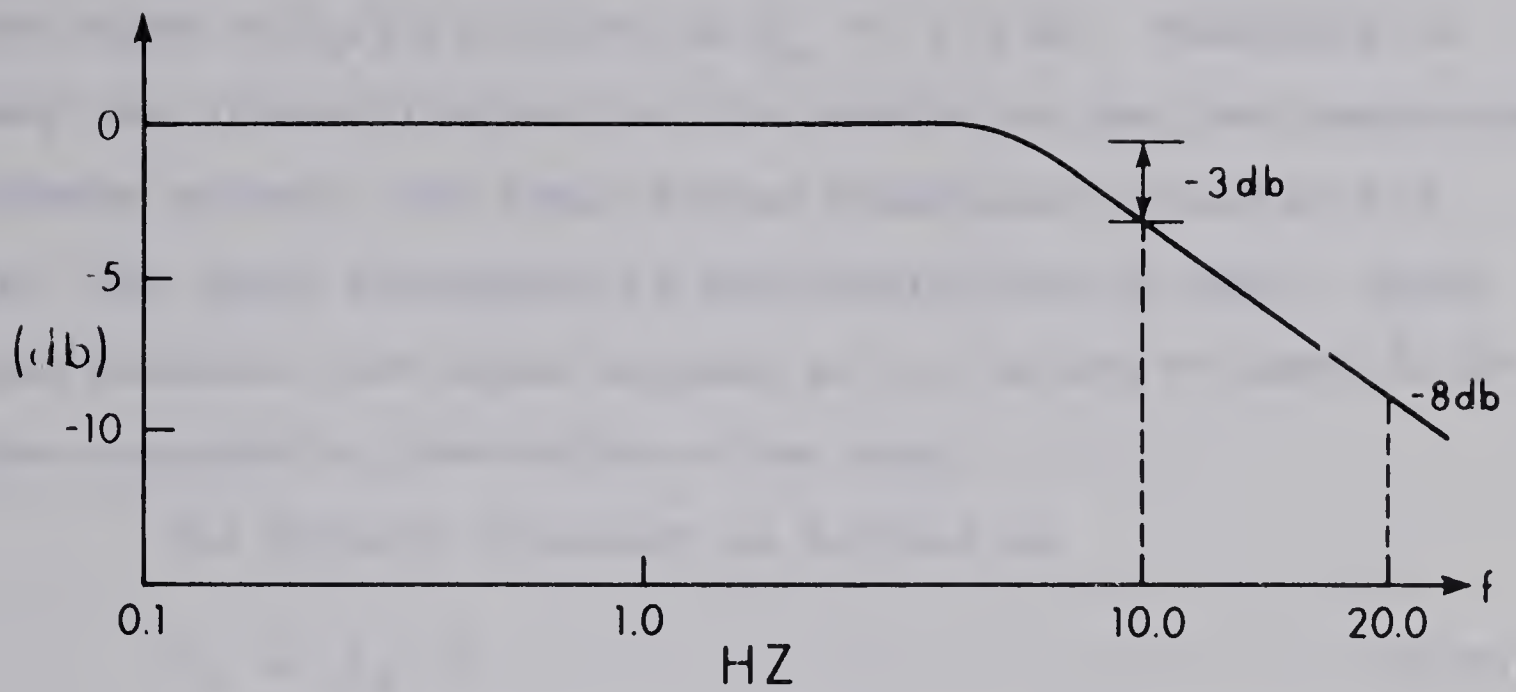


FIGURE 2.4 NOISE SPECTRUM OF SERVOMEX R.G. 77
RANDOM NOISE GENERATOR

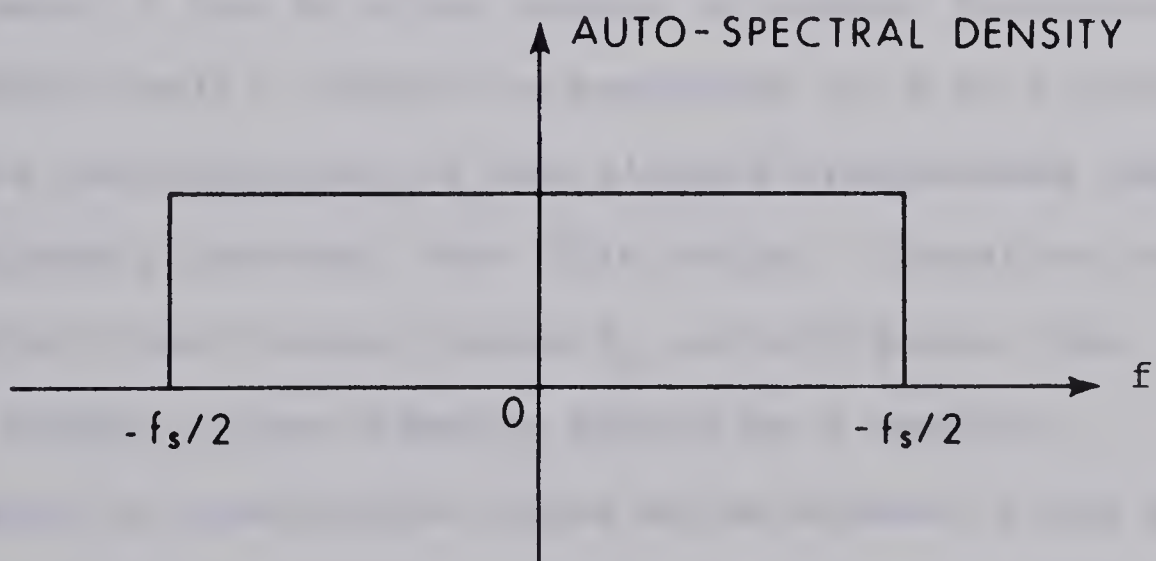


FIGURE 2.5 AUTO-SPECTRAL DENSITY OF BANDLIMITED
WHITE NOISE AFTER SAMPLING

If the sampling rate of the system is 1.0 second, the value of $f_s/2$ is 0.5 Hz as $f_s = 1.0$ Hz. Therefore to keep the aliased frequencies from adding to the low frequencies already present, the lower filter breakpoint is set at 1.0 Hz. The upper breakpoint is arbitrarily set at 10Hz. Using this passband, the input signals at 0.5 Hz are at least 24 db down compared to the middle of the band.

The Nyquist frequency is defined as

$$f_c = f_s / 2 \quad (2.70)$$

For any frequency f where $0 \leq f \leq f_c$, the higher frequencies that aliase with f are:⁷

$$(2f_c \pm f), (4f_c \pm f), \dots, (2Nf_c \pm f), \dots \quad (2.71)$$

Each frequency f has an equal number of higher frequencies aliased with itself. Since the bandwidth of 0 to 0.5 Hz is narrow, the contributions of the aliased frequencies should be approximately constant over this range. Therefore since the amount of input signal below f_c is negligible, the resultant output of the sampler should be a usable approximation to bandlimited white noise between 0 and 0.5 Hz.

2-6 TEST PROCEDURE ON THE HYBRID SIMULATION

The noise generated by the method outlined in Section 2-5 is applied to the system shown in FIG. 2.3. Experimental runs of 100 seconds duration were taken. A run of 100

seconds allows the measurement of at least one complete cycle of all frequencies above 0.01 Hz.

Every additional decade below this requires an extra factor of 10 in the length of the run. The additional accuracy obtained could be offset by difficulties with the total drift and overall accuracy of the analog components.

The hybrid test procedure is as follows:

1. The minimum time deadbeat controller is tested with a unit step to verify the deadbeat response.
2. This controller is then tested in the presence of noise.
3. The extended deadbeat controllers are tested with a unit step to verify the deadbeat response.
4. These extended deadbeat controllers are tested in the presence of noise.

Whenever a controller was tested, 7 to 10 runs of 100 seconds in length were taken. The mean of these runs was calculated along with the 95% confidence limits on the mean. These results are presented graphically in Chapter 3. The interpretation of these results and a proposed prediction criterion for the extended controllers are also given in Chapter 3. The hybrid system errors are common to both the minimum time and extended controllers and therefore the errors tend to cancel each other when the ratio of the noise transfers is taken.

CHAPTER III

HYBRID SIMULATION RESULTS AND THE DERIVATION OF A DESIGN CRITERION

In this chapter, the results and conclusions obtained from the hybrid computer simulation described in Chapter 2 are presented. The results obtained from the hybrid simulation are presented graphically and then some preliminary conclusions are stated. Some details of two criteria considered are then presented.

The final section contains the reasoning behind and the derivation of the proposed design criterion. The controllers derived by use of this criterion are then compared with the controllers which minimize the noise transfer of the system when tested on the hybrid.

3-1 SYSTEMS TESTED BY HYBRID SIMULATION

Using the methods outlined in Sections 2-3 and 2-4, controllers with one extra term were calculated by iterating e_1 . The plants considered were:

1. $G_1(s) = 1/s(s+1)$ with $T = 1$ second
2. $G_1(s) = 1/s(s+3)$ with $T = 1$ second
3. $G_1(s) = 1/s(s+1)$ with $T = 0.5$ second
4. $G_2(s) = 1/s(s+1)(s+2)$ with $T = 1$ second.

The noise transfers measured for the systems containing each controller are plotted versus e_1 which is the iterated variable. All noise transfers of the systems using the extended controllers are normalized with respect to the noise transfer of the system using the appropriate minimum time controller.

Controllers extended by two terms were also considered for $G_1(s) = 1/s(s+1)$ with $T = 1$ second. In this case, e_1 and e_2 were the unknown variables. To simplify the problem of calculating these controllers, e_2 was fixed at 0.50 and then e_1 was iterated. The normalized noise transfers for this system are then plotted versus e_1 with $e_2 = 0.50$.

3-2 GRAPHICAL RESULTS AND SUMMARY

As stated in Chapter 2, the mean noise transfer for a given system was the average of 7 to 10 test runs. The 95% confidence limits were also calculated for this mean value. For each minimum time controller, the mean and confidence limits of the noise transfer for the system are listed in Table 3.1.

FIG. 3.1 to FIG. 3.5 present the normalized noise transfers for the systems containing the extended controllers. The normalized noise transfer was calculated by dividing the mean noise transfer obtained using the extended controllers by the mean obtained using the appropriate minimum time controller. The 95% confidence limits were then calculated around the normalized mean.

This method of presentation is used to show more clearly the actual improvement due to the extended controllers.

<u>Plant</u>	<u>T</u> <u>Seconds</u>	Mean Noise Transfer Measured On <u>Hybrid (%)</u>	95% Confidence Limits On Mean	
			<u>Lower</u>	<u>Upper</u>
$\frac{1}{s(s+1)}$	1.0	88.686	83.359	94.013
$\frac{1}{s(s+3)}$	1.0	70.982	64.853	77.112
$\frac{1}{s(s+1)}$	0.5	146.18	139.47	152.89
$\frac{1}{s(s+1)(s+2)}$	1.0	95.876	84.398	107.36

TABLE 3.1 NOISE TRANSFERS AND CONFIDENCE LIMITS FOR
SYSTEMS CONTAINING THE MINIMUM TIME CONTROLLERS
MEASURED ON THE HYBRID

A summary of the results shown in FIG. 3.1 to FIG. 3.5 is given in Table 3.2. The results are given as the range of e_1 which yields the minimum normalized noise transfer and the range of the normalized noise transfers corresponding to the e_1 values.

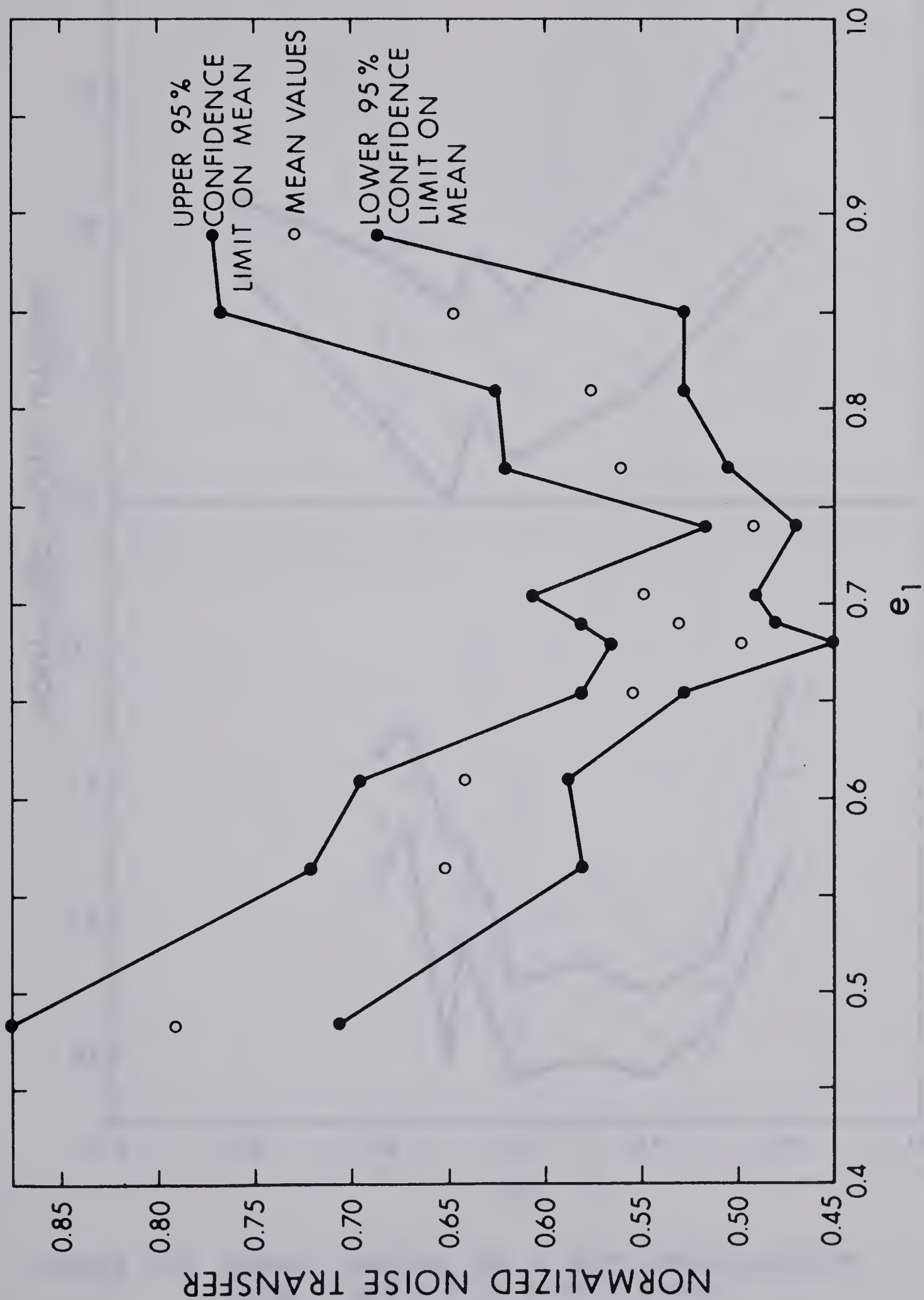


FIGURE 3.1 HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 1)$ WITH $T=1$ SEC.

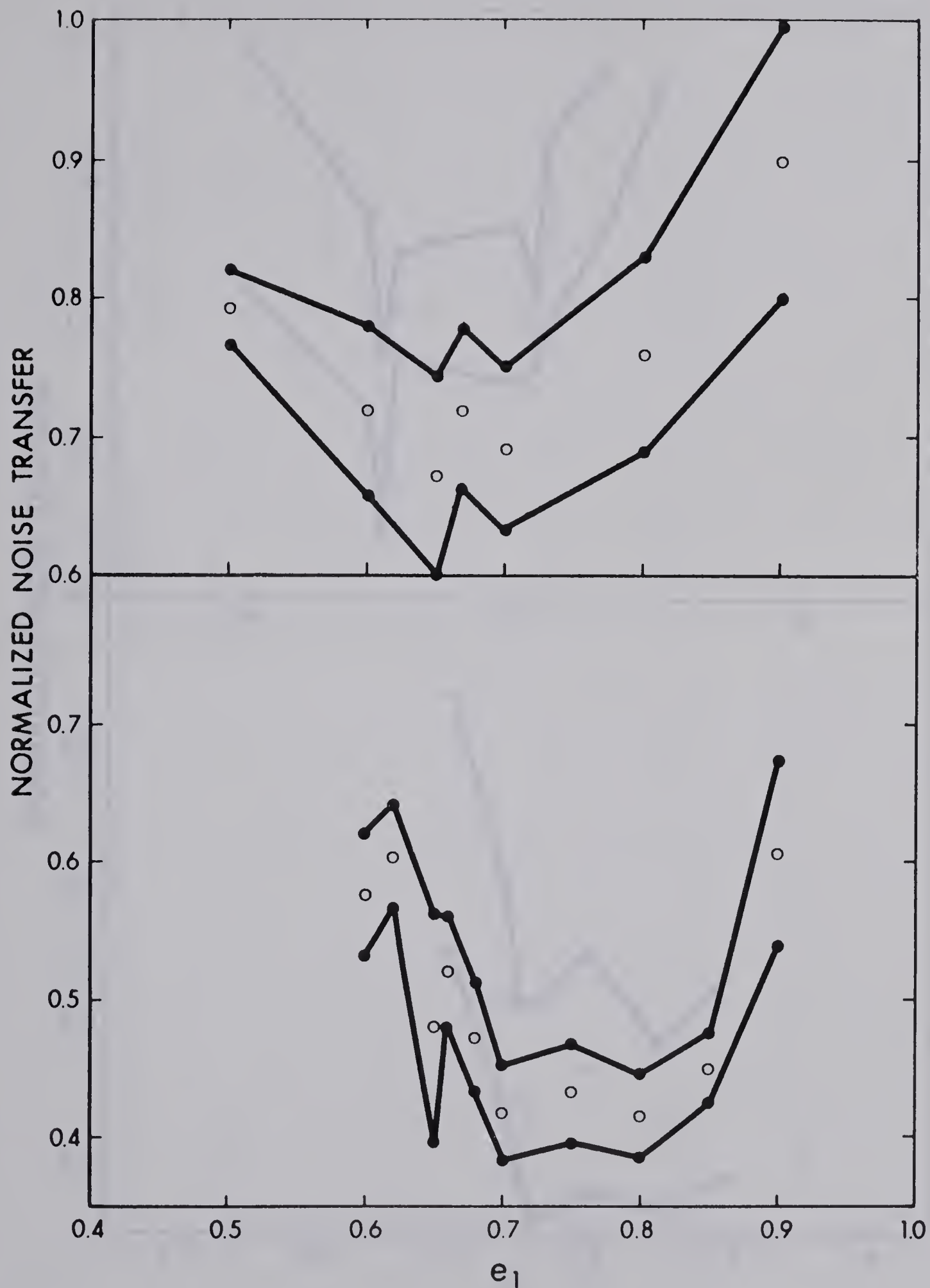


FIGURE 3.2 HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 3)$ WITH $T=1$ SEC.

FIGURE 3.3 HYBRID RESULTS FOR 3 TERM CONTROLLERS OF $G_1(s) = 1/s(s + 1)$ WITH $T=0.5$ SEC.

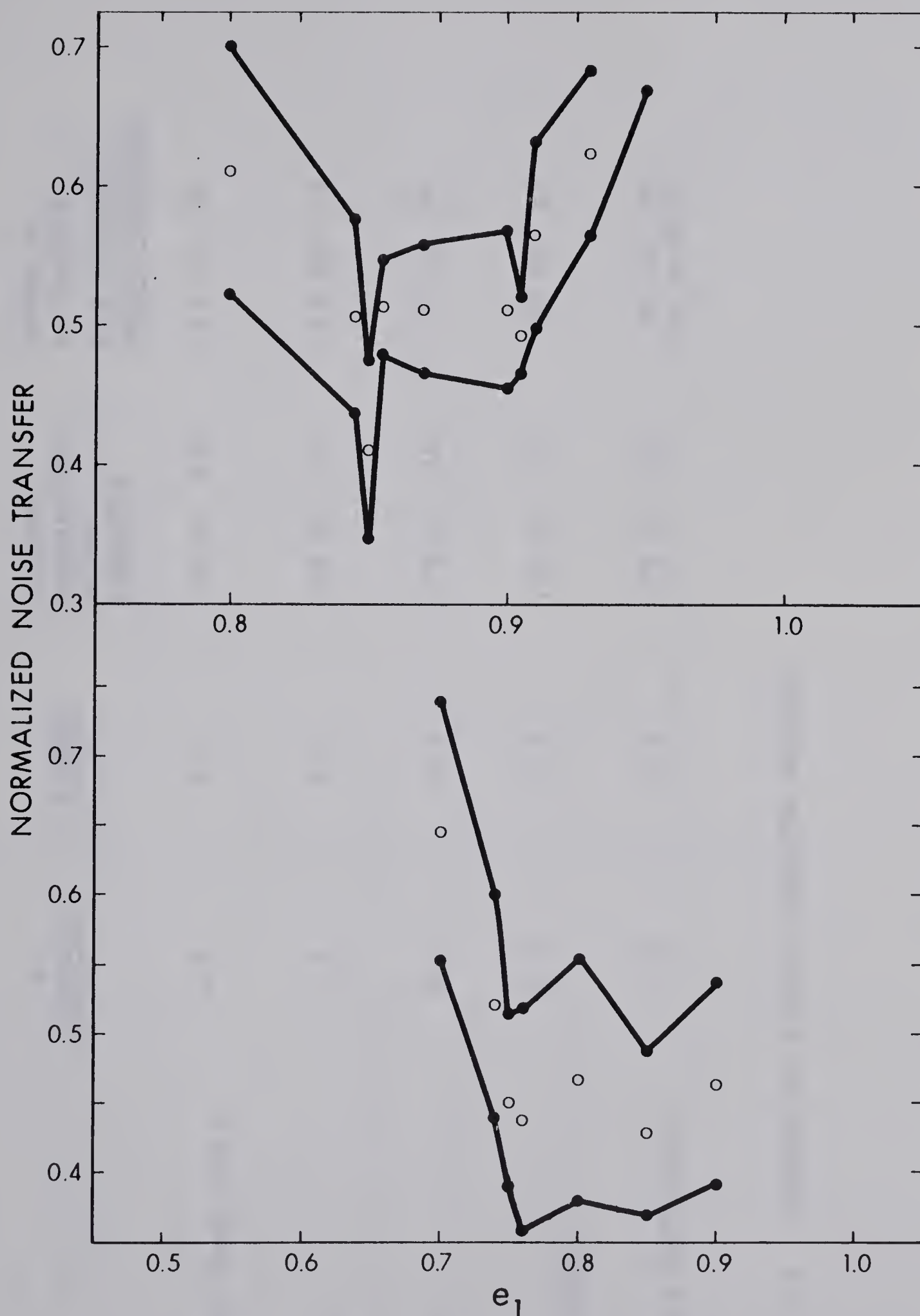


FIGURE 3.4 HYBRID RESULTS FOR 4 TERM CONTROLLERS OF $G_2(s) = 1/s(s+1)(s+2)$ WITH $T=1$ SEC.

FIGURE 3.5 HYBRID RESULTS FOR 4 TERM CONTROLLERS OF $G_1(s) = 1/s(s+1)$ WITH $T=1$ SEC. AND $e_2=0.5$

<u>PLANT</u>	<u>T (SECONDS)</u>	<u>FIGURE</u>	<u>MINIMUM NORMALIZED NOISE TRANSFER</u>	<u>RANGE OF e₁ VALUES FOR MINIMUM NOISE TRANSFER</u>
$\frac{1}{s(s+1)}(3 \text{ Term } D(z))$	1.0	3.1	.45 to .60	.67 to .75
$\frac{1}{s(s+3)}$	1.0	3.2	.60 to .77	.64 to .70
$\frac{1}{s(s+1)}$	0.5	3.3	.38 to .48	.70 to .80
$\frac{1}{s(s+1)(s+2)}$	1.0	3.4	.40 to .57	.82 to .90
$\frac{1}{s(s+1)}(4 \text{ Term } D(z))$	1.0	3.5	.35 to .55	.75 to .88 (e ₂ = 0.5)

TABLE 3.2 SUMMARY OF HYBRID COMPUTER RESULTS

\underline{m}	COEFFICIENTS OF BEL(0)	COEFFICIENTS OF BEL(T)	COEFFICIENTS OF BEL(2T)	COEFFICIENTS OF BEL(3T)
0	(h_0)	0	0	0
1	$(h_1 - e_1 h_0)$	(h_0)	0	0
2	$(h_2 - e_1 h_1 - e_2 h_0 + e_1^2 h_0)$	$(h_1 - e_1 h_0)$	(h_0)	0
3	$(h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + e_1^2 h_1 + 2e_1 e_2 h_0 - e_1^3 h_0)$	$(h_2 - e_1 h_1 - e_2 h_0 + e_1^2 h_0)$	$(h_1 - e_1 h_0)$	(h_0)

TABLE 3.3 COEFFICIENTS OF BEL (mT) FOR THE VARIOUS CYCLES OF THE DIGITAL CONTROLLER

From the summary in Table 3.2, a number of preliminary conclusions may be drawn:

1. It is possible to minimize the noise transfer of a system containing an extended controller with a particular value of e_1 .
2. The improvement is significant considering the added controller complexity when only one term is added. Comparing the results in FIG. 3.1 and FIG. 3.5 does not indicate if the improvement obtained by two extra terms is significant.
3. Generally the range of values of e_1 over which a minimum may occur is relatively wide.

Considering these results, it seems worthwhile to study this method of noise improvement in more detail.

3-3 CONSIDERATIONS INVOLVED IN THE DEVELOPMENT OF A DESIGN CRITERION FOR AN EXTENDED CONTROLLER

The results given in Section 3-2 were obtained experimentally. This method of finding the minimum noise transfer extended controllers is time consuming. A quicker method would be a design criterion to predict the minimum noise transfer extended controllers.

As stated in the introduction, the application of the criterion to a particular plant should be as practical as possible. This indicates that the criterion should be relatively simple in form. The simplest form of criterion

would be in terms of the plant parameters with no direct reference to the noise input.

If the input noise is white, this assumption is possible. Stationary white noise has a constant auto-spectral density and may be assumed to have no cross-correlation with a unit step. Providing that the noise is at least a reasonable approximation to white noise, it should be possible to neglect it in the derivation.

Several criteria were proposed and tested. The predicted results of each case were considered to verify the validity of the proposal.

3-4 CRITERIA CONSIDERED

The first method considered did not exclude the noise input as suggested in Section 3-3. This approach was based on the work by Tou¹ and corrected by Jury et al.¹⁰⁻¹³ All calculations of the optimum controllers in the presence of noise are carried out in the z -domain. This means the calculations are not simple in form. Since the method involves the inverse z -transformation, it is difficult to write the equations as part of a digital computer program.

Tou's method also requires a specified model response to be incorporated in the calculations. No such model can be predicted for the extended deadbeat controllers. Because of these difficulties, Tou's method is not considered further.

Another method considered is a "straight line" approximation. This means that the deadbeat response

obtained using the extended controller is given by an output which has the minimum deviation, at the sampling instants, from a straight line from $c[0] = 0$ to $c[nT] = 1.0$. The values of the "straight line" output at the sampling instants are: $c_{SL}[iT] = \frac{i}{n}$; $i = 1, 2, \dots, n-1$. Thus the errors for the "straight line" at the sampling instants are:

$$e_{iSL} = (1 - i/n); \quad i = 1, 2, \dots, n-1.$$

For the extended controllers, the deviations from the "straight line" approximation are given by:

$$\begin{aligned} \text{Deviation at } i^{\text{th}} \text{ sampling instant} &\equiv \text{DEV}_{iSL} = \\ &(e_{iSL} - e_i) \quad ; \quad i = 1, 2, \dots, n-1 \end{aligned}$$

Therefore the sum of these deviations must be minimized to yield the predicted controller.

This method is considered as it should have the smallest change in the equivalent gain of the controller from one period to the next.

The proposed criterion was applied to some of the plants. It was found to be a useful approximation in some cases, but the results in general were not satisfactory. Some instances when this is a useful approximation are presented in Chapter 7.

The failure of the "straight line" approximation to give general results indicates a more complicated criterion is needed. A more general criterion is derived in Section 3-5.

3-5 DERIVATION OF THE DESIGN CRITERION

In the system shown in FIG. 1.1, the plant parameters and sampling rate are assumed to be fixed. This leaves the parameters of the extended controller as the only variables. These must then be calculated so as to minimize the system noise transfer.

Since the plant is linear, the output of the system at any sampling instant may be related to the output of the controller at that instant. This relationship may be expressed as follows:

$$c[mT] = c_G[mT] \text{ BE2}[mT] \quad (3.1)$$

where $c_G[mT]$ is the combined equivalent gain of the plant and zero-order hold in the m^{th} sampling period.

This expression is arrived at by consideration of the discrete state equations of the system. In these equations, the plant is replaced by a matrix of values which vary with time but are constant over any one sampling period.

Using (3.1), the mean square error of the system, as evaluated at the sampling instants, becomes:

$$\overline{[c[mT]]^2} = \overline{[c_G[mT] \text{ BE2}[mT]]^2} \quad (3.2)$$

Rewriting (1.47) as a difference equation yields the controller output at the m^{th} sampling instant as:

$$\begin{aligned}
 \text{BE2}[mT] &= \sum_{k=0}^m (h_k) \text{BE1}[mT - kT] + \\
 &\quad \sum_{k=1}^m (e_k) \text{BE2}[mT - kT]
 \end{aligned} \tag{3.3}$$

where

$$0 \leq m \leq n \tag{3.4}$$

Expanding (3.3) for various values of m yields the following relationships.

For $m = 0$:

$$\text{BE2}[0] = (h_0) \text{BE1}[0] \tag{3.5}$$

For $m = 1$:

$$\begin{aligned}
 \text{BE2}[T] &= ((h_0) \text{BE1}[T] + (h_1) \text{BE1}[0]) - \\
 &\quad ((e_1) \text{BE2}[0])
 \end{aligned} \tag{3.6}$$

For $m = 2$:

$$\begin{aligned}
 \text{BE2}[2T] &= ((h_0) \text{BE1}[2T] + (h_1) \text{BE1}[T] + \\
 &\quad (h_2) \text{BE1}[0]) - ((e_1) \text{BE2}[T] - \\
 &\quad (e_2) \text{BE2}[0])
 \end{aligned} \tag{3.7}$$

For $m = 3$:

$$\begin{aligned}
 BE2[3T] = & ((h_0) BE1[3T] + (h_1) BE1[2T] + \\
 & (h_2) BE1[T] + (h_3) BE1[0]) - \\
 & ((e_1) BE2[2T] + (e_2) BE2[T] + \\
 & (e_3) BE2[0]) \quad (3.8)
 \end{aligned}$$

This expansion can be similarly extended for m up to n .

These expressions may be rewritten entirely in terms of the input $BE1[mT]$ in the following fashion.

For $m = 0$:

$$BE2[0] = (h_0) BE1[0] \quad (3.5)$$

For $m = 1$, substituting (3.5) into (3.6) yields in simplified form:

$$BE2[T] = (h_1 - e_1 h_0) BE1[0] + (h_0) BE1[T] \quad (3.9)$$

For $m = 2$, substituting (3.5) and (3.9) into (3.7) and simplifying yields:

$$\begin{aligned}
 BE2[2T] = & (h_2 - e_1 h_1 - e_2 h_0 + e_1^2 h_0) BE1[0] + \\
 & (h_1 - e_1 h_0) BE1[T] + (h_0) BE1[2T] \quad (3.10)
 \end{aligned}$$

For $m = 3$, substitute (3.5), (3.9) and (3.10) into (3.8) and simplify to obtain:

$$\begin{aligned}
 BE2[3T] = & (h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + e_1^2 h_1 + \\
 & 2e_1 e_2 h_0 - e_1^3 h_0) BE1[0] + (h_2 - e_1 h_1 - \\
 & e_2 h_0 + e_1^2 h_0) BE1[T] + (h_1 - e_1 h_0) \\
 & BE1[2T] + (h_0) BE1[3T]
 \end{aligned} \tag{3.11}$$

As shown in Table 3.3, the controller inputs $BE1[0]$ to $BE1[3T]$ are acted upon by the same set of controller coefficients. This set of coefficients is delayed for each input $BE1[mT]$ by mT as compared to $BE1[0]$. It is therefore possible to conclude that the net effect of the controller over all n periods is the same for $BE1[0]$ as for any other input.

Rewrite (3.5), (3.9), (3.10) and (3.11) considering only the terms with respect to $BE1[0]$ as follows:

$$BE2[0] \Big|_{BE1[0]} = (h_0) BE1[0] \tag{3.5}$$

$$BE2[T] \Big|_{BE1[0]} = (h_1 - e_1 h_0) BE1[0] \tag{3.12}$$

$$\begin{aligned}
 BE2[2T] \Big|_{BE1[0]} = & (h_2 - e_1 h_1 - e_2 h_0 + e_1^2 h_0) \\
 & BE1[0]
 \end{aligned} \tag{3.13}$$

$$\left. \text{BE2}[3T] \right|_{\text{BE1}[0]} = (h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + e_1^2 h_1 + 2e_1 e_2 h_0 - e_1^3 h_0) \quad (3.14)$$

If the net controller effect on any $\text{BE1}[mT]$ is constant, then it should only be necessary to minimize the net effect of the controller on $\text{BE1}[0]$. This corresponds to minimizing the effects of the inputs to the controller at all sampling instants. Since the error signal $\text{BE1}[mT]$ contains the noise signal, it is implied that the noise transfer through the controller is minimized by this calculation.

This means that the "weighting sequence" $\text{BE2}[mT]$ of the extended controller is minimized.

Expressing (3.1) in discrete form yields:

$$\left. c[mT] \right|_{\text{BE1}[0]} = c_G[mT] \left. \text{BE2}[mT] \right|_{\text{BE1}[0]} \quad (3.15)$$

The mean square error due to $\text{BE1}[0]$ becomes:

$$\overline{\left[\left. c[mT] \right|_{\text{BE1}[0]} \right]^2} = \overline{\left[c_G[mT] \left. \text{BE2}[mT] \right|_{\text{BE1}[0]} \right]^2} \quad (3.16a)$$

Ignoring the constant, (3.16a) becomes:

$$\overline{\left[\left. c[mT] \right|_{\text{BE1}[0]} \right]^2} = \overline{\left[\left. \text{BE2}[mT] \right|_{\text{BE1}[0]} \right]^2} \quad (3.16b)$$

where the net mean-square error of the controller to the input BE1[0] is defined as:

$$\overline{\left[\text{BE2 [mT]} \middle| \text{BE1 [0]} \right]^2} = \left(\frac{1}{\text{BE1 [0]}} \right)^2 \sum_{m=0}^n (\text{BE2 [mT]} \middle| \text{BE1 [0]})^2 \quad (3.16c)$$

Therefore to minimize the net effect of the controller on BE1[0], (3.16c) must be minimized.

Substituting (3.5), (3.12), (3.13) and (3.14) into (3.16c) yields the following simplified form:

$$\begin{aligned} \overline{\left[\text{BE2 [mT]} \middle| \text{BE1 [0]} \right]^2} &= (h_0)^2 + (h_1 - e_1 h_0)^2 + \\ &\quad (h_2 - e_1 h_1 - e_2 h_0 + e_1^2 h_0)^2 + \\ &\quad (h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + e_1^2 h_1 + \\ &\quad 2e_1 e_2 h_0 - e_1^3 h_0)^2 + \dots + \\ &\quad \left(\frac{1}{\text{BE1 [0]}} \right)^2 (\text{BE2 [nT]} \middle| \text{BE1 [0]})^2 \end{aligned} \quad (3.17)$$

This sum in (3.17) is the net controller mean-square error due to BE1[0]. Henceforth this term will be referred to by the abbreviation NCE. This abbreviation is defined as:

NCE \equiv NET CONTROLLER MEAN-SQUARE ERROR

$$\text{TO BE1[0]} = \overline{\left[\text{BE2[mT]} \middle| \text{BE1[0]} \right]^2} \quad (3.18)$$

The proposed design criterion, which when minimized yields the predicted extended controllers, is obtained by combining (3.17) and (3.18) into the following form:

$$\begin{aligned} \text{NCE} = & (h_0)^2 + (h_1 - e_1 h_0)^2 + (h_2 - e_1 h_1 - e_2 h_0 + \\ & e_1^2 h_0)^2 + (h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + e_1^2 h_1 + \\ & 2e_1 e_2 h_0 - e_1^3 h_0)^2 + \dots + \left(\frac{1}{\text{BE1[0]}} \right)^2 \\ & \left(\text{BE2[nT]} \middle| \text{BE1[0]} \right)^2 \end{aligned} \quad (3.19)$$

This criterion is defined entirely in terms of the parameters of the controller. These parameters, in turn, are functions of the plant parameters and the sampling rate of the system only.

Therefore even if a particular plant has a peak or dip at some frequency, the extended controller predicted by NCE should still give the minimum noise transfer for the bandlimited white noise. This conclusion is based on the fact that NCE may be written entirely as a function of the plant parameters and the sampling rate. Therefore, any dips or peaks in the frequency response of the plant will be

considered in the calculation of the predicted extended controller.

3-6 COMPARISON OF CRITERION RESULTS AND HYBRID RESULTS

For the five systems considered at the start of this chapter, the criterion of section 3-5 is used to find the values of e_1 which yield the minimum values of NCE. These results are compared in Table 3.4 to the hybrid results listed in Table 3.2.

From the results in Table 3.4, it is apparent that acceptable agreement exists between the hybrid and NCE values of e_1 . In all cases, the value of e_1 predicted by NCE lies in the middle of the range of e_1 values from the hybrid.

These results indicate that this proposed criterion is worth additional consideration. To further test this criterion, more simulation results for the predicted controllers are required.

In Chapter 4, an experimental simulation using a digital computer is developed to supply these additional results.

<u>PLANT</u>	<u>T (SECONDS)</u>	<u>e₁ FROM HYBRID RESULTS</u>	<u>e₁ FROM PREDICTED FROM NCE</u>
$\frac{1}{s(s+1)}(3 \text{ Term } D(z))$	1.0	.67 to .75	.7205
$\frac{1}{s(s+3)}$	1.0	.64 to .70	.6556
$\frac{1}{s(s+1)(s+2)}$	0.5	.70 to .80	.7427
$\frac{1}{s(s+1)(s+2)}$	1.0	.82 to .90	.8429
$\frac{1}{s(s+1)}(4 \text{ Term } D(z))$	1.0	.75 to .88 (e ₂ = 0.5)	.85 (e ₂ = 0.5)

TABLE 3.4 COMPARISON OF HYBRID AND NCE RESULTS

CHAPTER IV

DIGITAL COMPUTER SIMULATION

In this chapter, the digital computer simulation used to test the controllers calculated by NCE is outlined. The hybrid simulation results, with which the predictions by the design criterion are compared in Section 3-6, are for only two types of plants. The digital simulation is used to test the four types of plants listed in Chapter I. The results of these simulations are given in Chapters 5 to 7.

The digital simulation is used instead of the hybrid simulation for the following reasons:

1. The controllers can be tested more rapidly.
2. A larger combination of plants and controller extensions may be considered.
3. Repeatable results are obtainable with precisely defined noise sources.
4. The simulation is free of the component drift present in the hybrid simulation.
5. Accuracy of test results up to 16 significant figures is possible if desired. The actual calculation accuracy is 5 significant figures.

In the first section, the method of simulation is outlined. Then the test procedure followed is given. The

noise sources used in the test program are discussed in the final section of this chapter.

4-1 METHOD OF DIGITAL SIMULATION

Fortran IV is the programming language used on an IBM 360/67 computer. The systems are simulated by use of the discrete state equations derived in Chapter 1.

Various noise sources are simulated by the random number generation programs available in the IBM Scientific Subroutine Package. Since the mean-square error of the system is calculated only at the sampling instants, the input noise need only be defined at the sampling instants to yield the system noise transfer value.

The mean-square sampled error may be calculated theoretically for systems with a noise contaminated input. To perform this calculation, it is necessary to know the auto-spectral and cross-spectral densities of the unit step and noise inputs. It is also necessary to take the inverse z -transform of the system transfer function. The variable s is then replaced in the resultant function by $-s$ and the z -transform is then taken. Then a contour integral is evaluated.

This calculation is expressed in equation form by Tou¹ as:

$$\overline{e^2(nT)} = \frac{1}{2\pi j} \oint_{\Gamma} \phi_{ee}(z) z^{-1} dz \quad (4.1)$$

$$\text{where } \phi_{ee}(z) = M(z^{-1}) M(z) \phi_{nn}(z) \quad (4.2)$$

$$\text{and where } M(z) = C(z)/R(z)$$

In this calculation, the following assumptions are made:

1. There is no correlation between the noise and the unit step input.
2. The model system response is set equal to the system response obtained using the extended controller being tested. This removes the terms from (4.2) which contain the auto-spectral density of the unit step $\phi_{uu}(z)$.
3. $\phi_{nn}(z)$ is the auto-spectral density of the input noise. For white noise, $\phi_{nn}(z)$ is a constant.

These calculations are very difficult to program on the digital computer due to the transforms and contour integration required. Therefore the alternate approach of using random number sets to simulate the white noise input is used.

This method yields only an approximation to white noise as the experiment is time limited. The validity of any results obtained by this method may be evaluated if the accuracy of the approximation is known. These random number approximations are discussed in Section 4-3.

The Fortran program consists of the following sections:

1. A group of algorithms to calculate the various controllers.
2. A system simulation to verify that deadbeat to a unit step is satisfied.
3. A system simulation to test the system in the presence of noise.
4. An algorithm to calculate the mean-square error for the minimum time controllers and the normalized mean-square errors of the extended controllers.

The block diagram of the simulation given by FIG. 4.1 incorporates these sections.

The diagram of the computer simulation shown in FIG. 4.1 is greatly simplified. All details are omitted. The program for the sections which verify deadbeat and calculate the noise transfers is given in Appendix B.

The algorithms for the calculation of the various predicted extended controllers are given in the appropriate chapters.

4-2 DIGITAL COMPUTER SIMULATION TEST PROCEDURE

As shown in FIG. 4.1, there are three types of controllers to be considered for a given system. The computer simulation package can be run in two forms. These are:

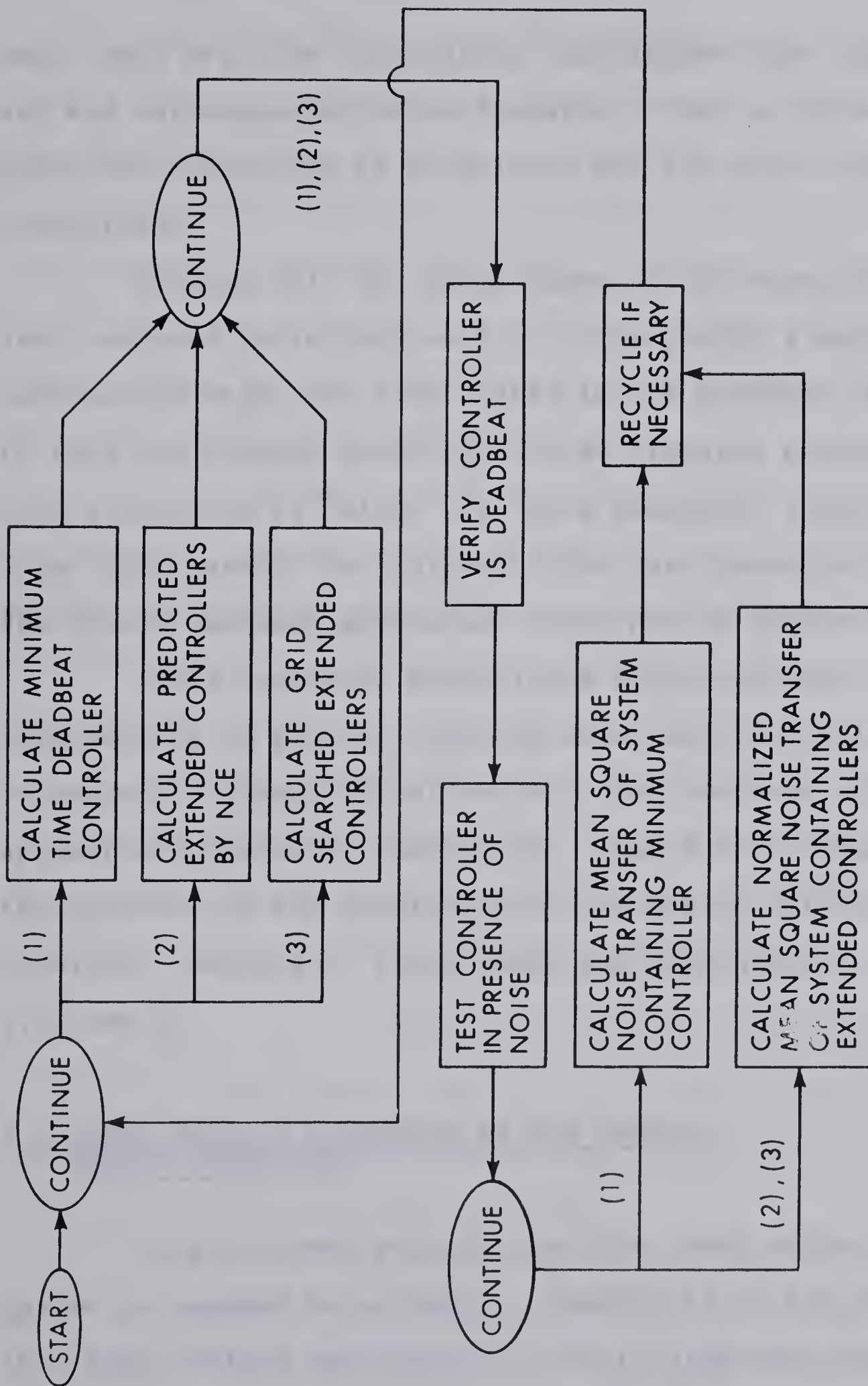


FIGURE 4.1 SIMPLIFIED BLOCK DIAGRAM OF DIGITAL COMPUTER SIMULATION

- a. Branches (1) and (2) compared.
- b. Branches (1) and (3) compared.

Both forms begin by calculating the minimum time controller and its corresponding noise transfer. Then in form (a), the predicted controller is calculated and its noise transfer is normalized.

In form (b), the error terms of the extended controller are iterated as in Section 2-4. This yields a series of controllers which are then tested in the presence of noise to find the minimum normalized noise transfer controller. This controller is called the "grid searched" controller. This "grid search" is similar to the test procedure used in the hybrid computer simulation described in Chapters 2 and 3.

"Grid searched" controllers are calculated for various combinations of plants, sampling rates and controller extensions. These controllers are then compared with the appropriate predicted controller. From these comparisons, the accuracy of the prediction by the design criterion is obtained. Results of these tests are presented in Chapters 5, 6 and 7.

4-3 NOISE SOURCES SIMULATED BY THE RANDOM NUMBER GENERATORS

It was stated earlier that the input noise for the system is assumed to be white. However it is not possible in a time limited experiment to obtain wide-band white noise. It is possible to generate noise which is white over some

limited range of frequencies. This is called bandlimited white noise.

After sampling, the noise bandwidth is reduced to 0 to $\omega_s/2$ where ω_s is the sampling frequency. If the noise in this frequency range is white, the distribution of frequencies is uniform from 0 to $\omega_s/2$. This occurs if the input noise before sampling is white and tightly low-passed from 0 to $\omega_s/2$. Then the output of the sampler is pure bandlimited white noise. In this case, the bandlimited noise referred to is actually low-pass noise in all cases.

It is useful to calculate the Fourier coefficients of the sampled noise sources simulated by the random number sets used in the digital simulation. The distribution of these coefficients may be compared to the ideal bandlimited white noise frequency distribution obtained after sampling.

Using a Fortran subroutine called PS301A, the real and imaginary Fourier coefficients are calculated for the noise sources by the Fast Fourier method. The frequencies for which these coefficients are calculated are $0, (\frac{\omega_s}{2L}), 2(\frac{\omega_s}{2L}), 3(\frac{\omega_s}{2L}), \dots, (\frac{L-1}{2L})\omega_s$ where L is the total number of points in the random number set. At each point, the magnitude of the Fourier coefficient is calculated in decibels from the real and imaginary values.

Once these magnitudes are known, a comparison may be made. The mean of these magnitudes for each noise source is calculated. Then the variance and deviation of the magnitudes around the mean are evaluated for each source.

Since low-pass white noise has a uniform distribution over 0 to $\omega_s/2$, its variance and deviation around the mean are zero. Therefore the lower the values of the variance and deviation for each noise source, the better the approximation.

The Scientific Subroutine Package of Fortran IV contains two random number generation programs. These are called:

1. RANDU
2. GAUSS

The subroutine RANDU generates uniformly distributed random numbers between 0 and 1.0. In order to make this sequence useful, 0.5 is subtracted from all values when calculated. This results in a uniformly distributed set of numbers with amplitudes between -0.5 and + 0.5.

The second subroutine GAUSS generates a set of random numbers with a specified variance around a specified mean. The amplitude distribution of these numbers is Gaussian. For all Gaussian sets used, the mean is 0, but the variances differ.

By varying initializing coefficients in the subroutines, different sets of random numbers are available. The length of the set of random numbers is variable up to 10^{29} numbers without repetition of values. For normal testing, the sets generated contain 256 or 1024 numbers. The reasons for these lengths are given in Chapter 5.

Five different noise sources are used. These are labeled:

1. GAUSS2A
2. GAUSS2
3. GAUSS3

which are generated by the GAUSS subroutine and

4. RANDUA
5. RANDU

which are generated by the RANDU subroutine.

Table 4.1 lists the Fourier statistics calculated for each noise source by the subroutine PS301A.

<u>NOISE SOURCE</u>	<u>LENGTH OF NUMBER SET</u>	<u>FOURIER COEFFICIENT STATISTICS</u>	
		<u>MEAN (decibels)</u>	<u>DEVIATION ABOUT MEAN (decibels)</u>
GAUSS2A	1024	7.6053	5.4287
GAUSS2	256	1.4664	5.2050
GAUSS3	256	1.7052	5.3203
RANDUA	1024	16.746	5.7010
RANDU	256	10.618	5.6861

TABLE 4.1 FOURIER COEFFICIENT STATISTICS FOR RANDOM
NUMBER NOISE SOURCES

In Table 4.1, only the deviation is given as the variance is simply the square of the deviation. From this table, GAUSS2 appears to be the best approximation as it has the lowest deviation. In all cases, the deviation is substantially greater than zero however.

CHAPTER V

CONTROLLERS EXTENDED ONE TERM

The results of the digital simulation tests of the controllers with one extra term are presented in this chapter. To simplify the presentation, a controller extended by one term is referred to as a PlC. This stands for Plus One Term Controller.

The calculation algorithm for the PlC is derived from NCE in the first section. In the following sections, the test results for the four plants listed in Chapter 1 are presented.

The results for these plants consist of two parts.

1. The controllers predicted by NCE are presented for various sets of system parameters.
2. The predicted controllers are compared to the "grid searched" controllers using the various noise sources.

The responses of the minimum time deadbeat controllers and the predicted PlCs to ramp and acceleration inputs are compared in the second last section. In the last section, some conclusions obtained from the test results are given.

5-1 ALGORITHM USED IN THE DIGITAL SIMULATION TO CALCULATE THE PREDICTED PLCs

The four plants defined by (1.4) to (1.7) were tested using digital simulation. In order to test these plants, the PLCs must be calculated. The method presented in Section 2-4 is used as the basis of the algorithm.

Consider the plant $G_2(s)$. In Table 2.2(b) the coefficients of the $D(z)$ are expressed in terms of e_1 . Since NCE is derived in Chapter 3 in terms of the coefficients of the $D(z)$, it is possible to rewrite NCE as a function of e_1 .

The calculation algorithm yields the predicted value of e_1 using NCE. This allows the predicted PLC to be calculated. Although $G_2(s)$ is used as an example during this derivation, the algorithm derived is general.

The digital controller coefficients listed in Table 2.2(b) may be expressed as follows:

$$h_0 = a_1 e_1 + a_2 \quad (5.1)$$

$$h_1 = a_3 e_1 + a_4 \quad (5.2)$$

$$h_2 = a_5 e_1 + a_6 \quad (5.3)$$

$$h_3 = a_7 e_1 + a_8 \quad (5.4)$$

$$e_2 = b_1 e_1 + b_2 \quad (5.5)$$

$$e_3 = b_3 e_1 + b_4 \quad (5.6)$$

where the coefficients a_1 to a_8 and b_1 to b_4 for $G_2(s)$ are defined in Table 5.1(a).

The criterion proposed in Chapter 3 for extended controllers with $n \leq 3$ is

$$\begin{aligned} \text{NCE} = & h_0^2 + (h_1 - h_0 e_1)^2 + (h_2 - h_1 e_1 - h_0 e_2 + \\ & h_0 e_1^2)^2 + (h_3 - h_2 e_1 - h_1 e_2 - h_0 e_3 + h_1 e_1^2 - \\ & h_0 e_1^3 + 2h_0 e_1 e_2)^2. \end{aligned} \quad (5.7)$$

(5.7) may be rewritten in the following form:

$$\text{NCE} = \sum_{i=0}^n \text{NCE}_i \quad (5.8)$$

where

$$\text{NCE}_0 = (h_0)^2 \quad (5.9)$$

$$\text{NCE}_1 = (h_1 - h_0 e_1)^2 \quad (5.10)$$

$$\text{NCE}_2 = (h_2 - h_1 e_1 - h_0 e_2 + h_0 e_1^2)^2 \quad (5.11)$$

$$\begin{aligned} \text{NCE}_3 = & (h_3 - h_2 e_1 - h_1 e_2 - h_0 e_3 + h_1 e_1^2 - \\ & h_0 e_1^3 + 2h_0 e_1 e_2)^2 \end{aligned} \quad (5.12)$$

For $G_2(s)$, the PLC has $n = 3$. By substituting (5.1) to (5.6) into (5.9) to (5.12), NCE may be written as a function of e_1 . Define NCE for a PLC as NCE_1 which is a function of e_1 . Using this definition yields (5.9) as:

$$NCE_{10} = a_1^2 e_1^2 + 2a_1 a_2 e_1 + a_2^2 \quad (5.13)$$

(5.10) as:

$$\begin{aligned} NCE_{11} = & a_1^2 e_1^4 - 2a_1 a_{11} e_1^3 + (a_{11}^2 - 2a_1 a_4) \\ & e_1^2 + 2a_4 a_{11} e_1 + a_4^2 \end{aligned} \quad (5.14)$$

(5.11) as:

$$\begin{aligned} NCE_{12} = & a_1^2 e_1^6 + 2a_1 a_{21} e_1^5 + (a_{21}^2 + 2a_1 a_{22}) \\ & e_1^4 + 2(a_1 a_{23} + a_{21} a_{22}) e_1^3 + (a_{22}^2 + \\ & 2a_{21} a_{23}) e_1^2 + 2a_{22} a_{23} e_1 + a_{23}^2 \end{aligned} \quad (5.15)$$

(5.12) as:

$$\begin{aligned} NCE_{13} = & a_1^2 e_1^8 - 2a_1 a_{31} e_1^7 + (a_{31}^2 - 2a_1 a_{32}) e_1^6 + \\ & 2(a_{31} a_{32} - a_1 a_{33}) e_1^5 + (a_{32}^2 + 2a_{31} a_{33} - \\ & 2a_1 a_{34}) e_1^4 + 2(a_{31} a_{34} + a_{32} a_{33}) e_1^3 + (a_{33}^2 + \\ & 2a_{32} a_{34}) e_1^2 + 2a_{33} a_{34} e_1 + a_{34}^2 \end{aligned} \quad (5.16)$$

where

$$\begin{aligned}
 a_{11} &= a_3 - a_2 \\
 a_{21} &= a_2 - a_3 - a_1 b_1 \\
 a_{22} &= a_5 - a_4 - a_1 b_2 - a_2 b_1 \\
 a_{23} &= a_6 - a_2 b_2 \\
 a_{31} &= a_3 - a_2 + 2a_1 b_1 \\
 a_{32} &= a_4 - a_5 - a_3 b_1 - a_1 b_3 + 2(a_1 b_2 + a_2 b_1) \\
 a_{33} &= a_7 - a_6 + 2a_2 b_2 - a_3 b_2 - a_4 b_1 - a_1 b_4 - a_2 b_3 \\
 a_{34} &= a_8 - a_4 b_2 - a_2 b_4
 \end{aligned} \tag{5.17}$$

Substituting (5.13) to (5.16) into (5.8) yields the following simplified form:

$$NCE_1 = \sum_{i=0}^{2n+2} C_i e_1^i = \sum_{i=0}^n (NCE_{1i}) \tag{5.18}$$

where C_i is the coefficient of e_1^i generated by summing the coefficients of e_1^i in (5.13) to (5.16). Thus for $G_2(s)$, the criterion is an 8th order linear equation in e_1 as shown in (5.19).

$$NCE_1 = \sum_{i=0}^8 C_i e_1^i \tag{5.19}$$

This expression must be minimized in order to obtain the predicted value of e_1 . Because of the deadbeat restriction, e_1 for any valid controller must satisfy the following conditions:

$$e_1 \text{ is real} \quad (5.20a)$$

$$0 \leq e_1 \leq 1.0 \quad (5.20b)$$

One method for finding the minimum of an equation is differentiation. Differentiating (5.19) with respect to e_1 yields:

$$\frac{d}{de_1} (NCE_1) = \sum_{i=1}^8 i C_i e_1^{i-1} \quad (5.21)$$

Set (5.21) to zero:

$$\frac{d}{de_1} (NCE_1) = 0 = \sum_{i=1}^8 i C_i e_1^{i-1} \quad (5.22)$$

The roots of this equation yield the values of e_1 which correspond either to the local maxima or minima of the expression NCE_1 .

A polynomial root solving subroutine called POLRT is available in the Fortran Scientific Subroutine Package. This solves for the real and imaginary parts of all the roots of a polynomial of up to 36th order.

Any roots not satisfying the restrictions of (5.20)

n FOR PLC		$G_1(s) = \frac{1}{3} \frac{1}{s(s+\alpha)}$	$G_2(s) = \frac{1}{4} \frac{1}{s(s+\alpha)(s+\beta)}$
a ₁		$-\alpha^2 / (\alpha T + A - 1)$	$-1 / R7$
a ₂		$-a_1$	$-a_1$
a ₃		$(1 + A) a_2$	$(A + B + 1) / R7$
a ₄		$(\alpha / T (1 - A)) - a_3$	$-a_3 + 1 / (R7 + R8 + U)$
a ₅		$a_1 A$	$a_1 (AB + A + B)$
a ₆		$-a_5 - (\alpha A / T (1 - A))$	$-a_5 - (A + B) / (R7 + R8 + U)$
a ₇		-	$A (B) a_2$
a ₈		-	$A B (a_1 + 1 / (R7 + R8 + U))$
b ₁		$(1 - A - \alpha TA) / (\alpha T + A - 1)$	$R8 / R7$
b ₂		$-b_1 + (1 - A - \alpha TA) / \alpha T (1 - A)$	$-b_1 + (R8 + U) / (R7 + R8 + U)$
b ₃		-	$U / R7$
b ₄		-	$(U a_8) / (AB)$

TABLE 5.1(a) COEFFICIENTS OF EQUATIONS (5.1) TO (5.6) FOR THE PLCs OF G₁(s) AND G₂(s)

n FOR PlC	$\frac{G_3(s) = s+\beta/s(s+\alpha)}{3}$	$\frac{G_4(s) = 1/s(s^2+2\zeta\omega_n s + \omega_n^2)}{4}$
a ₁	- 1/B	- 1/A1
a ₂	-a ₁	-a ₁
a ₃	a ₂ (1 + A)	a ₁ (AC8 -1)
a ₄	-a ₃ + 1/(B + C)	-a ₃ + 1/(A1 + A2 + A3)
a ₅	a ₁ A	a ₁ (AC9 - AC8)
a ₆	-a ₅ - A/(B + C)	-a ₅ + a ₂ (AC8)
a ₇	-	a ₂ (AC9)
a ₈	-	AC9 (a ₁ + 1/(A1 + A2 + A3))
b ₁	C/B	a ₂ (A2)
b ₂	-b ₂ + C/(B + C)	-b ₁ + (A2 + A3)/(A1 + A2 + A3)
b ₃	-	a ₂ (A3)
b ₄	-	-b ₃ + A3/(A1 + A2 + A3)

TABLE 5.1(b) COEFFICIENTS OF EQUATIONS (5.1) TO (5.6) FOR THE PLCs OF G₃(s) AND G₄(s)

may be ignored. It was found that for every combination of α , β and T checked, only one root of (5.21) would satisfy (5.20). For these cases, the calculated value of e_1 corresponded to the only minimum of NCE_1 .

Similar calculations were performed for $G_1(s)$, $G_3(s)$ and $G_4(s)$. For all combinations of parameters attempted, only one value of e_1 was found in each case which satisfied (5.20). In each case this value of e_1 corresponded to the only minimum of NCE_1 .

The coefficients a_1 to a_8 and b_1 to b_4 are listed in Table 5.1(a) and 5.1(b) for $G_1(s)$ to $G_4(s)$. These coefficients are defined in terms of the appropriate constants from Chapter 1.

From these observations, a general algorithm for the calculation of the predicted PLC is as follows:

1. Determine n which is the order of the PLC to be calculated.
2. Calculate NCE_1 as shown in (5.18).

$$NCE_1 = \sum_{i=0}^{2n+2} C_i e_1^i \quad (5.18)$$

3. Differentiate (5.18) with respect to e_1 and set the result to zero.

$$\frac{d}{de_1} (NCE_1) = \sum_{i=1}^{2n+2} i C_i e_1^{i-1} = 0 \quad (5.23)$$

4. Solve for the roots of (5.23) and choose as e_1 the root which satisfies (5.20).

5. Substitute this value of e_1 into (5.1) to (5.6) to calculate the predicted minimum noise transfer PLC.

5-2 RESULTS FOR THE PLCs of $G_1(s) = 1/s(s+\alpha)$

The greatest amount of work was done for this plant as it was the first and simplest to be tested. Also any conclusions obtained from it may be applicable to later plants tested. The test procedure followed is outlined in Chapter 4.

The sampling rate T and the pole α are the only system parameters considered in NCE_1 . Therefore T was fixed and successive values of α were used in testing the plants.

The first set of data collected for $G_1(s)$ is obtained using the GAUSS2A and RANDUA noise sources. Both are 1024 sampling periods long. FIG. 5.1 shows the values of e_1 predicted by NCE_1 plotted versus α with $T = 1$ second. It is only necessary to present e_1 as all the other coefficients of the $D(z)$ can be calculated using the relationships given in Table 5.1.

FIG. 5.2 presents the normalized noise transfers of the systems using the predicted PLCs when the noise sources are GAUSS2A and RANDUA. The noise transfer is shown in all following graphs as a fraction. This means the noise transfer of the system containing the PLC is normalized to the noise transfer of the system containing the appropriate minimum

time deadbeat controller.

All calculations following are carried out for $T = 1$ second as the predicted values of e_1 are constant for constant values of the normalized constant αT . For example if $T = 10$ seconds and $\alpha = 0.1$, the e_1 predicted is the same as for $\alpha = 10$ and $T = 0.1$ second.

A grid search was then performed with both GAUSS2A and RANDUA. The number of values of α considered is less than for the predicted PLCs as the computer time to perform the grid search is much longer. Also a good indication of the agreement should be obtained by taking a reasonable cross-section of values. In order to further reduce the computer time required, the value of e_1 was only calculated to two decimal places. FIG. 5.3 presents the comparison between the predicted values of e_1 and the values of e_1 found by the grid searches.

The percentage differences between the predicted and grid searched PLCs are now considered. All the differences between the $D(z)$ error terms are calculated as a percentage of the predicted value. The normalized noise transfer differences are expressed as a percentage of the predicted noise transfer.

The noise transfer difference using GAUSS2A is - 1.5% at $\alpha T = 0.1$ and zero at $\alpha T = 96.0$. Using RANDUA, the difference is - 3.3% at $\alpha T = 0.1$ and zero at $\alpha T = 96.0$. The comparable differences in e_1 are - 4.4% and + 2.6% from the GAUSS2A grid search. Using RANDUA, the differences are

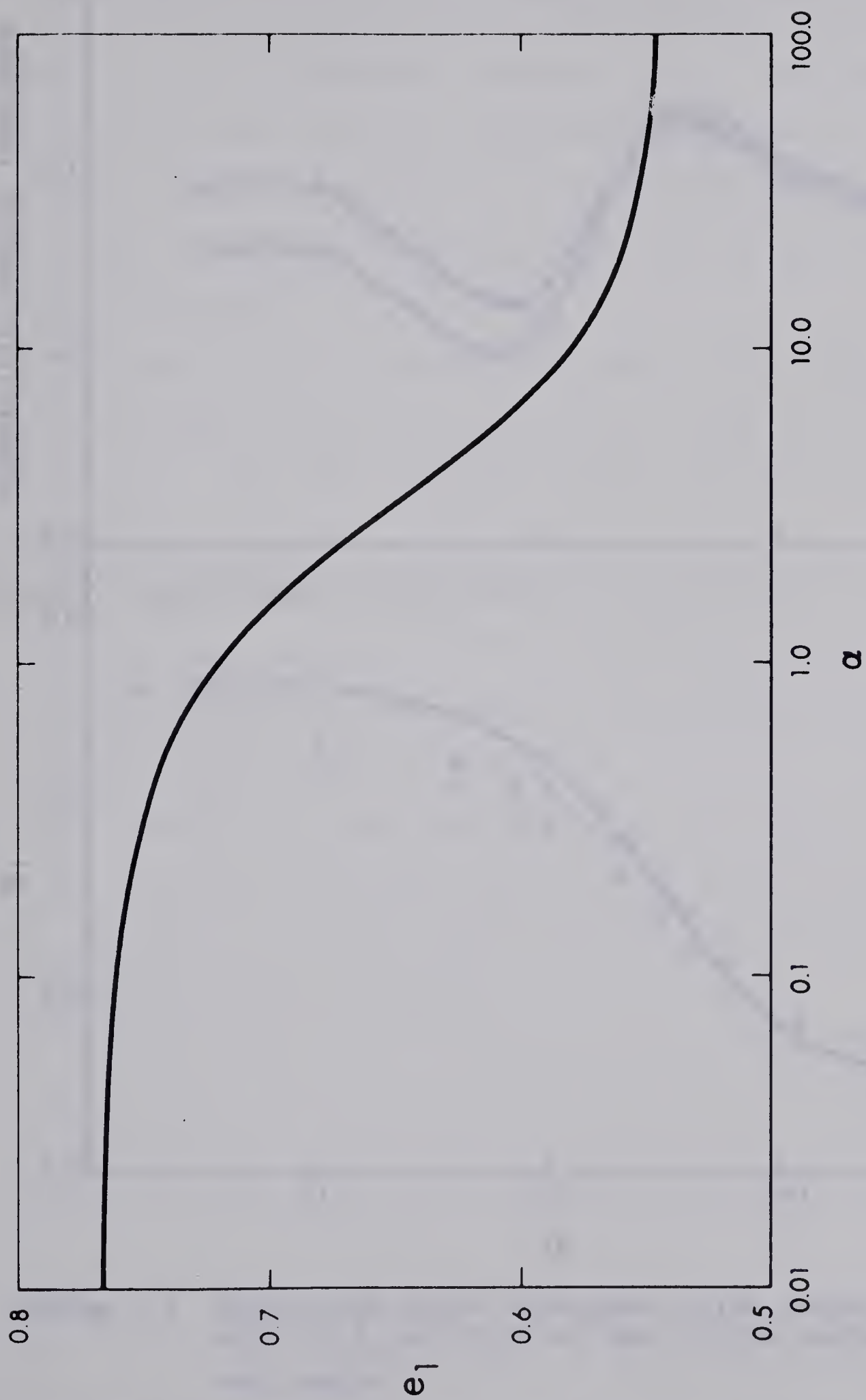


FIGURE 5.1 PREDICTED e_1 FOR PLCS OF $G_1(s)$ WITH $T=1$ SEC.

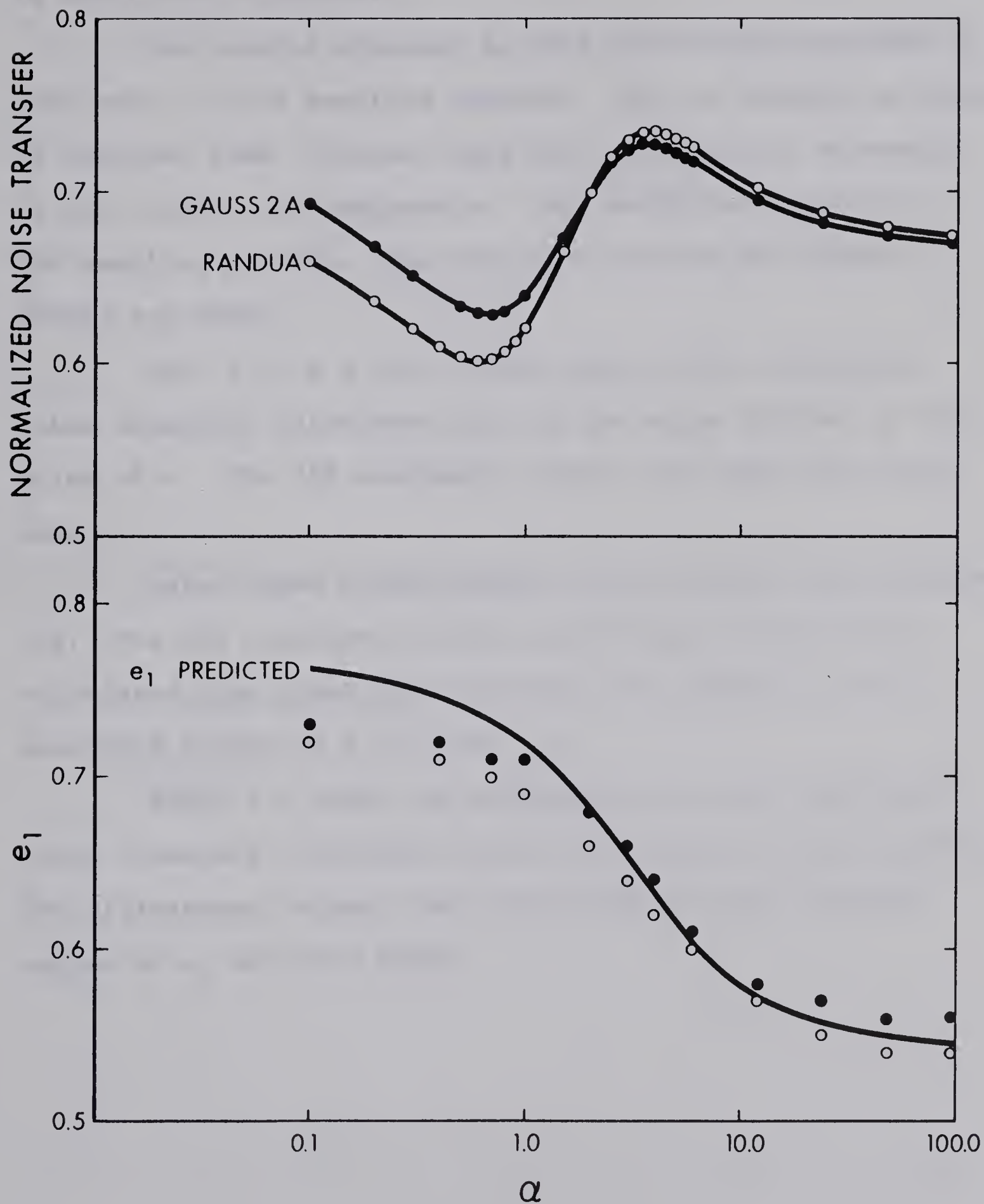


FIGURE 5.2 NORMALIZED NOISE TRANSFERS USING PREDICTED PLCs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2 AND RANDUA

FIGURE 5.3 PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2A AND RANDUA, FOR PLCs OF $G_1(s)$ WITH $T=1$ SEC.

- 5.5% at $\alpha T = 0.1$ and - 0.9% at $\alpha T = 96.0$. This agreement is considered acceptable.

The results obtained to this point were generated by test runs of 1024 sampling periods. This is lengthy in terms of computer time. Shorter runs were attempted to determine if the results are comparable. The length was reduced to 256 sampling periods. The new noise sources are GAUSS2, GAUSS3 and RANDU.

FIG. 5.4 is a plot of the mean of the normalized noise transfers calculated for all the noise sources at each value of α . The 95% confidence limits are shown with each mean.

Using these noise sources, grid searches were carried out. The 95% confidence limits of the mean values of e_1 calculated from these grid searches are compared to the predicted values of e_1 in FIG. 5.5.

Table 5.2 shows the differences for the normalized noise transfers calculated using the different noise sources. The differences between the calculated and grid searched values of e_1 are also shown.

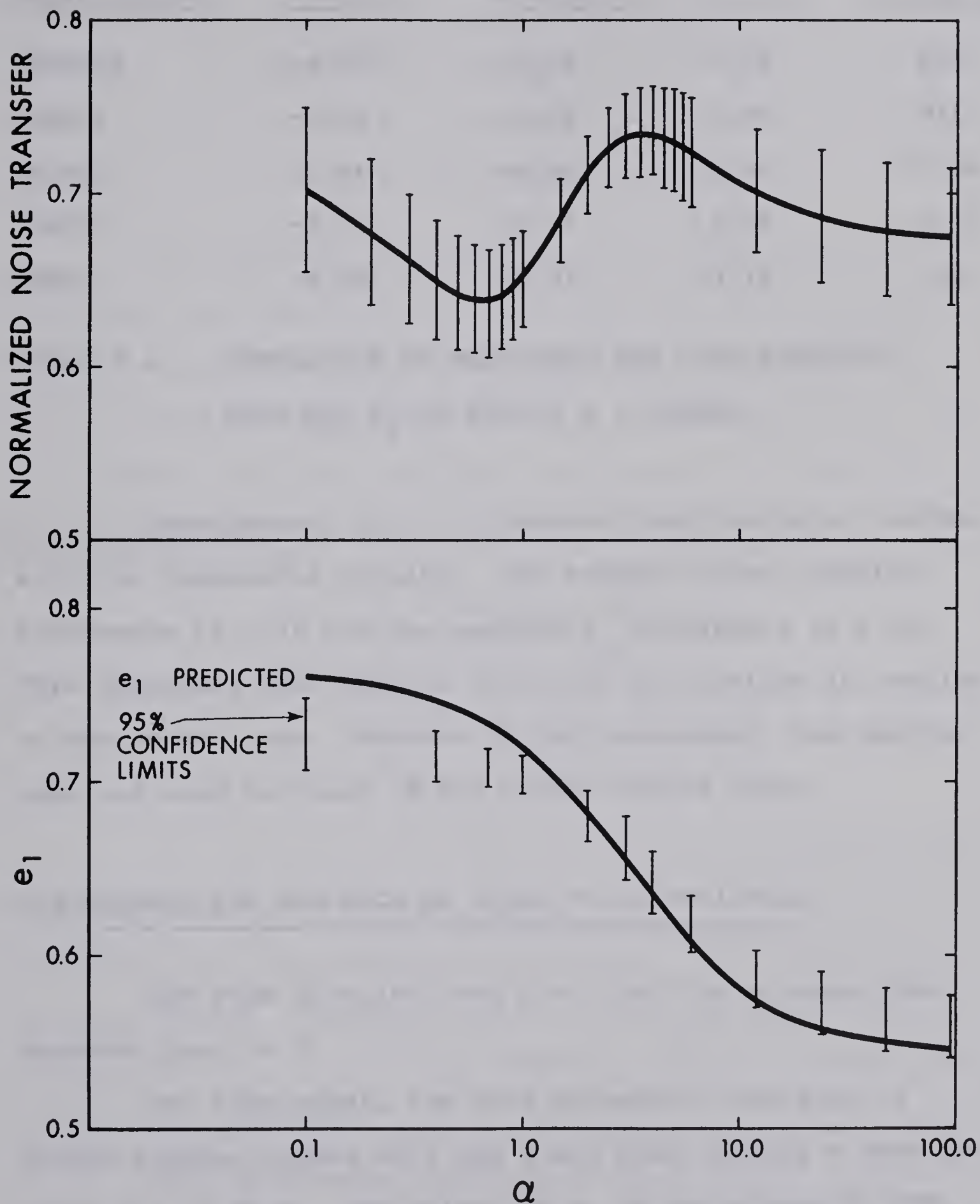


FIGURE 5.4 95% CONFIDENCE LIMITS ON MEAN NORMALIZED NOISE TRANSFER USING PlCs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 5.5 95% CONFIDENCE LIMITS ON MEAN VALUES OF GRID SEARCHED e_1 COMPARED TO PREDICTED e_1 FOR PlCs OF $G_1(s)$ WITH $T=1$ SEC.

<u>Noise Source</u>	<u>e_1</u> <u>Differences</u>		<u>Normalized Noise</u> <u>Transfer Differences</u>	
	<u>$\alpha = 0.1$</u>	<u>$\alpha = 96.0$</u>	<u>$\alpha = .1$</u>	<u>$\alpha = 96.0$</u>
GAUSS2A	-4.4%	+2.6%	-1.5%	NIL
RANDUA	-5.5%	-0.9%	-3.3%	NIL
GAUSS2	-1.6%	+6.4%	-0.2%	-0.3%
GAUSS3	-5.4%	+2.7%	-3.3%	-0.1%
RANDU	-6.8%	+0.9%	-3.7%	NIL

TABLE 5.2 COMPARISON OF PREDICTED AND GRID SEARCHED
PLCs FOR $G_1(s)$ WITH $T = 1$ SECOND

From Table 5.2, it is evident that the noise sources all give comparable results. The maximum noise transfer difference is 3.7% and the maximum e_1 difference is 6.8%. This indicates that shorter test runs are similar in results to the longer runs. Because of this agreement, the shorter runs are used for most of the plants tested later.

5-3 RESULTS FOR THE PLCs of $G_2(s) = 1/s(s+\alpha)(s+\beta)$

The PLCs of $G_2(s)$ have $n = 3$ and the minimum time deadbeat has $n = 2$.

For this plant, the test procedure consisted of fixing various values of β and T and then varying α over the range 0.1 to 96.0. The values of e_1 as calculated by NCE_1 are plotted for the following sets of parameters:

1. $\beta = 0.1$, $T = 1$ second in FIG. 5.6
2. $\beta = 1.0$, $T = 1$ second in FIG. 5.6
3. $\beta = 10.0$, $T = 1$ second in FIG. 5.7
4. $\beta = 100.0$, $T = 1$ second in FIG. 5.7
5. $\beta = 1.0$, $T = 10$ seconds in FIG. 5.7
6. $\beta = 1.0$, $T = 100$ seconds in FIG. 5.7

The normalized noise transfers for the systems containing the predicted PLCs were then calculated using GAUSS2A. This noise source was used when calculating the noise transfer of the predicted PLCs. The shorter test runs were used in the grid searches. The plants for which the predicted normalized noise transfer is plotted are:

1. $\beta = 0.1$, $T = 1$ second in FIG. 5.8
2. $\beta = 1.0$, $T = 1$ second in FIG. 5.8
3. $\beta = 10.0$, $T = 1$ second in FIG. 5.8
4. $\beta = 100.0$, $T = 1$ second in FIG. 5.8

As in Section 5.2, a grid search was carried out at selected values of the plant coefficients to compare the predicted and grid searched values of e_1 . The noise source used was GAUSS2. The plant used had $\beta = 1.0$ and $T = 1$ second with $\alpha = .1, 1.5, 12.0$ and 96.0 . FIG. 5.9 shows the values of e_1 compared.

The comparative results are always within 0.5% of each other for the noise transfer calculations. Also the agreement between the predicted and grid searched controllers

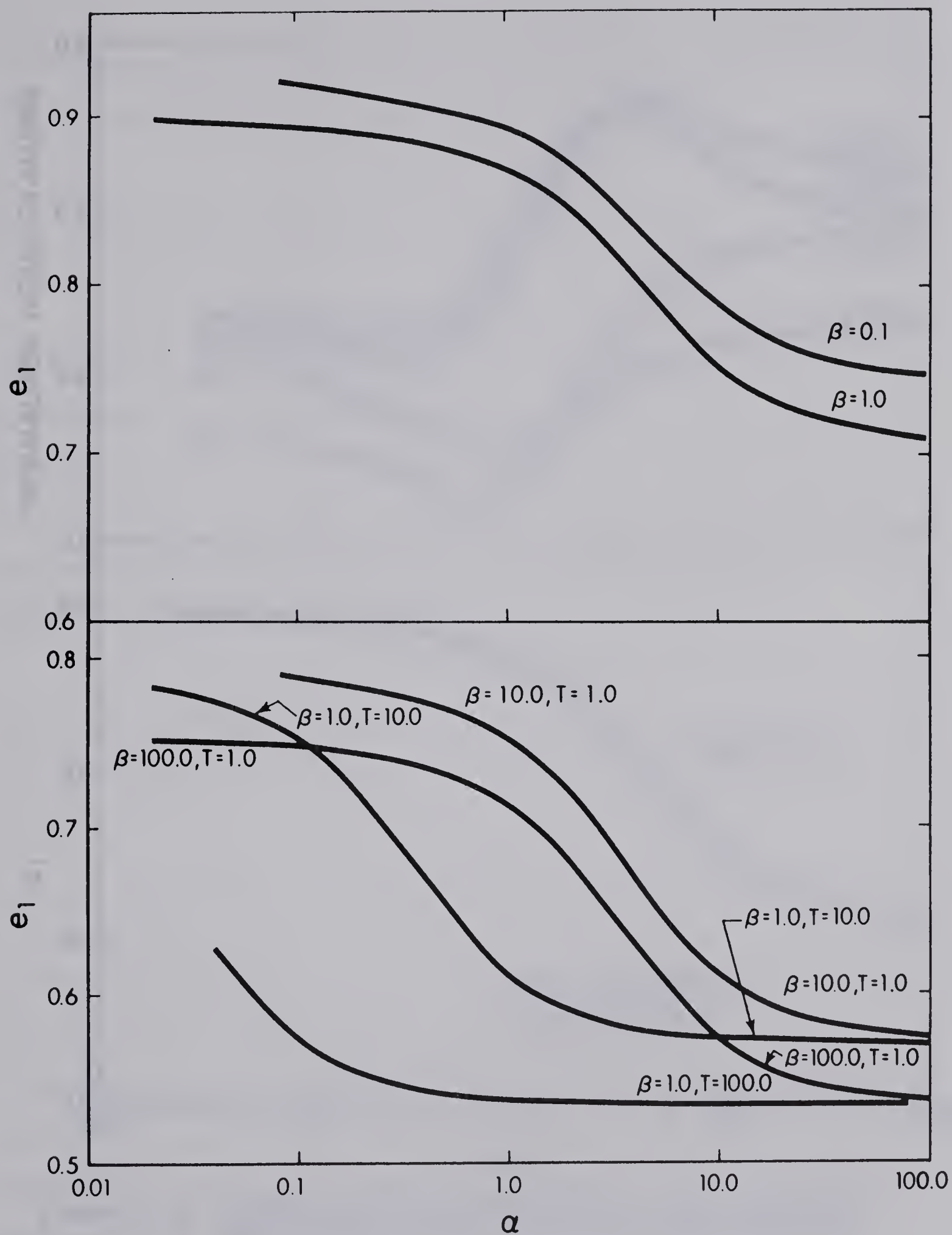


FIGURE 5.6 PREDICTED e_1 FOR PLCs OF $G_2(s)$ WITH $T=1$ SEC.

FIGURE 5.7 PREDICTED e_1 FOR PLCs OF $G_2(s)$ WITH $T=1$ & 10 SEC.

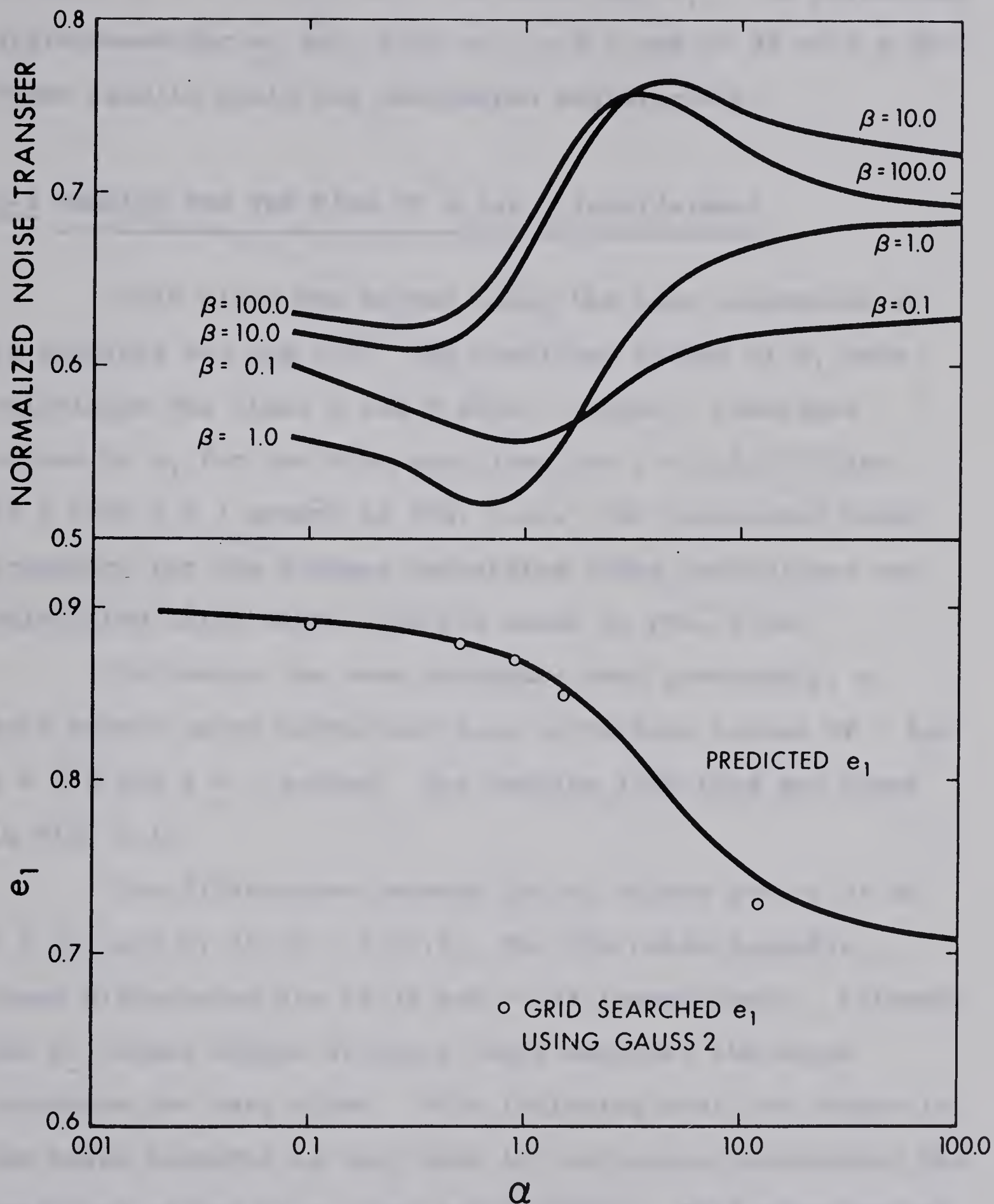


FIGURE 5.8 NORMALIZED NOISE TRANSFERS USING PREDICTED P1CS OF $G_2(s)$ WITH $T=1$ SEC. USING GAUSS2A

FIGURE 5.9 PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2, FOR P1CS OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.

is best at $\alpha = 0.1$ for both the noise and e_1 . The percentage differences for e_1 are -0.2% at $\alpha = 0.1$ and -0.3% at $\alpha = 96.0$. These results again are considered satisfactory.

5-4 RESULTS FOR THE PLCs OF $G_3(s) = (s+\beta)/s(s+\alpha)$

This plant was tested using the same procedures as in Sections 5-2 and 5-3. The predicted values of e_1 were calculated for fixed β and T with α varied. Predicted values of e_1 for the PLCs are given for $\beta = 2.0, 5.0$ and 50.0 with $T = 1$ second in FIG. 5.10. The normalized noise transfers for the systems containing these controllers are calculated using GAUSS2 and are shown in FIG. 5.11.

Following the same procedure used previously, a grid search using GAUSS2 was done at various values of α for $\beta = 5.0$ and $T = 1$ second. The results from this are shown in FIG. 5.12.

The differences between the e_1 values are -2.7% at $\alpha = 0.1$ and $+7.1\%$ at $\alpha = 96.0$. For the noise transfer, these differences are $+0.3\%$ and -0.4% respectively. Although the e_1 values differ by quite large margins, the noise transfers are very close. This indicates that the change in the noise transfer is very slow in the region surrounding the predicted values of e_1 when using GAUSS2. This experimental agreement is considered satisfactory.

For some combinations of α , β and T , these plants exhibit an overshoot in the deadbeat response to a unit step.

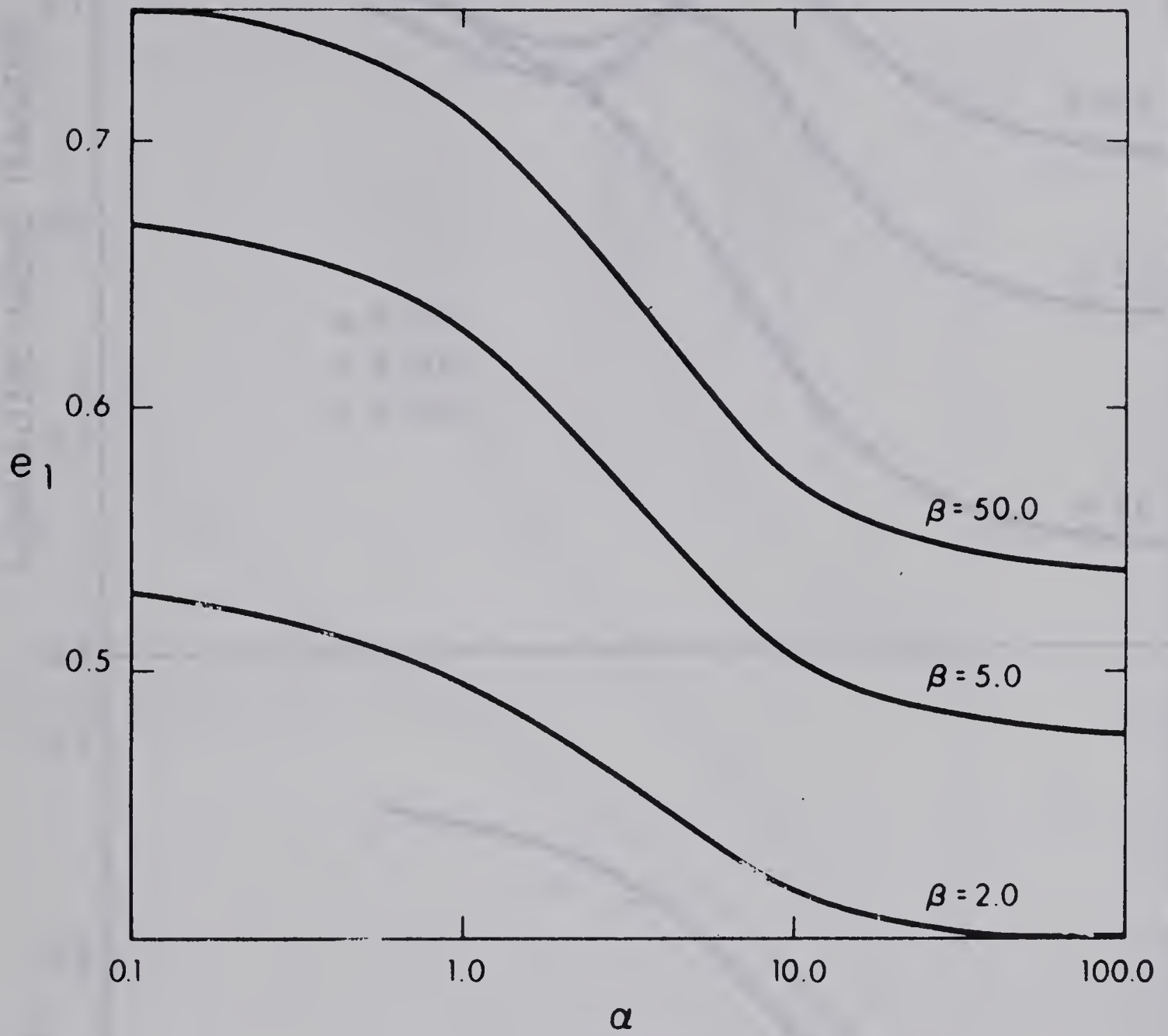


FIGURE 5.10 PREDICTED e_1 FOR PLCs OF $G_3(s)$ WITH $T=1$ SEC.

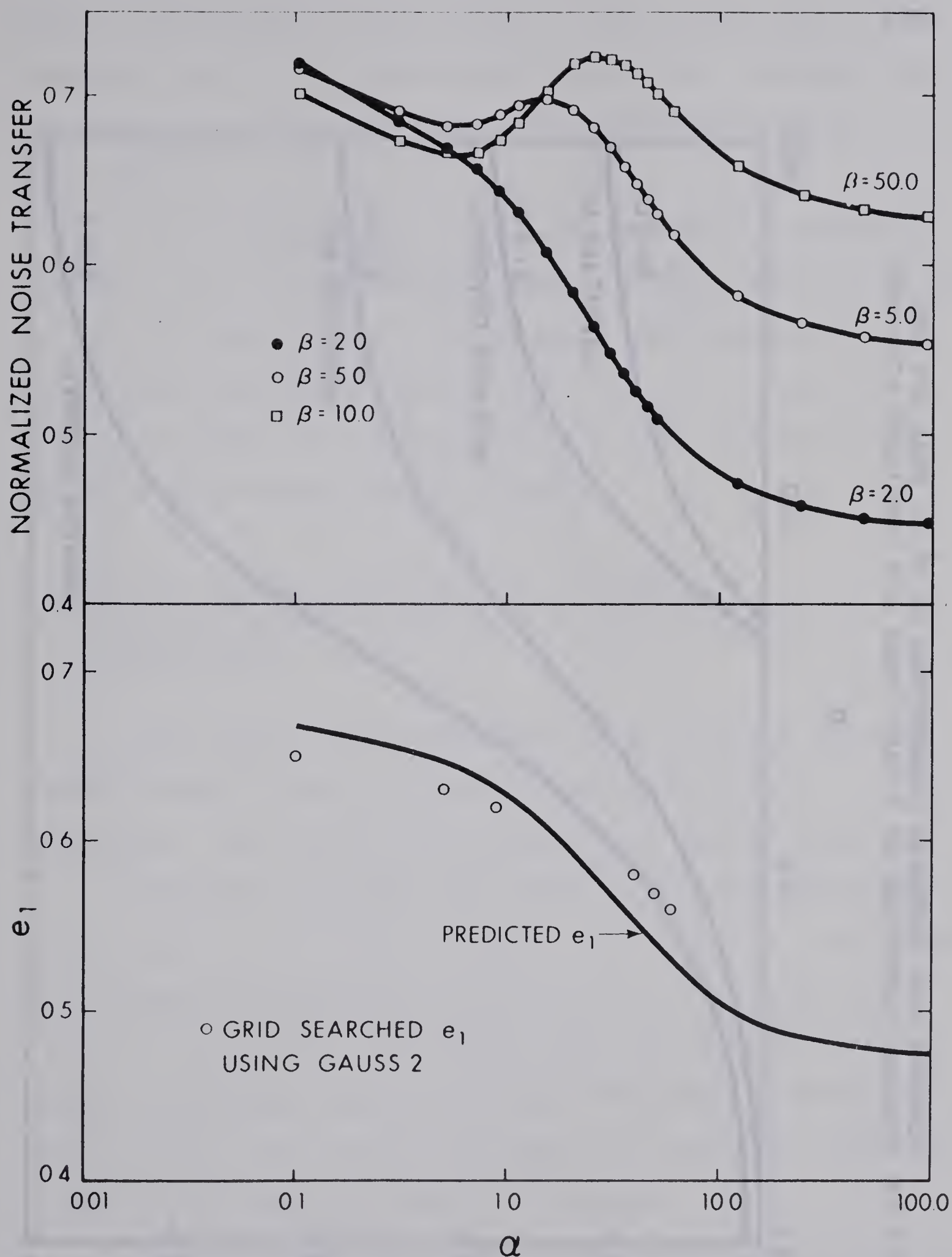


FIGURE 5.11 NORMALIZED NOISE TRANSFERS USING PREDICTED PLCS OF $G_3(s)$ WITH $T=1$ SEC. USING GAUSS2

FIGURE 5.12 PREDICTED e_1 AND GRID SEARCHED e_1 , USING GAUSS2, FOR PLCS OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.

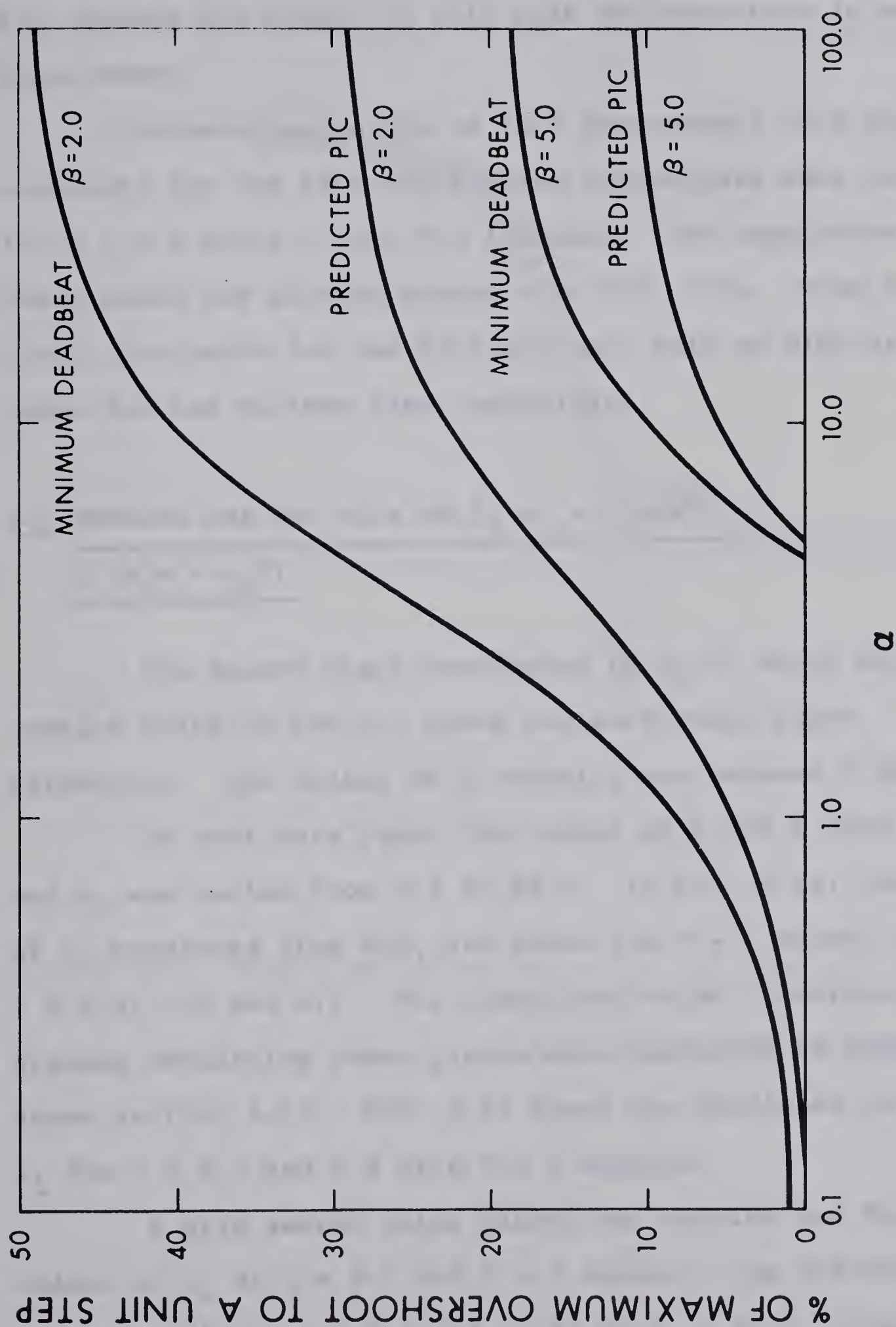


FIGURE 5.13 MAXIMUM OVERSHOOTS TO A STEP INPUT FOR PREDICTED PICs AND MINIMUM TIME DEADBEAT CONTROLLERS OF $G_2(s)$ WITH $T=1$ SEC.

This is illustrated in FIG. 1.2(b). The minimum time controller and the PLC both exhibit this peak. However the PLC reduces the height of this peak and therefore is an improvement.

As an illustration of this improvement, the deadbeat responses for the PLCs and minimum controllers were calculated for $\beta = 2.0$ and 5.0 with $T = 1$ second. The magnitudes of these peaks are plotted versus α in FIG. 5.13. From this graph, the peaks for the PLCs are only half as high as the peaks for the minimum time controllers.

5-5 RESULTS FOR THE PLCs OF $G_4(s) = 1/s(s^2 + 2\zeta\omega_n s + \omega_n^2)$

The fourth plant considered is $G_4(s)$ which may have complex poles in the s - plane for particular plant parameters. The values of ζ normally are between 0 and 1.

To test this plant the values of ζ and T were fixed and ω_n was varied from 0.1 to 96.0. In FIG. 5.14, the values of e_1 predicted from NCE_1 are shown for $T = 1$ second and $\zeta = 0.3, 0.5$ and 0.7 . The normalized noise transfers of the systems containing these plants when subjected to GAUSS2 are shown in FIG. 5.15. FIG. 5.15 shows the predicted values of e_1 for $\zeta = 0.3$ and 0.5 with $T = 5$ seconds.

A grid search using GAUSS2 was carried out for selected values of ω_n at $\zeta = 0.5$ and $T = 1$ second. The differences for e_1 are $+0.4\%$ at $\omega_n = 0.1$ and -2.6% at $\omega_n = 96.0$. The

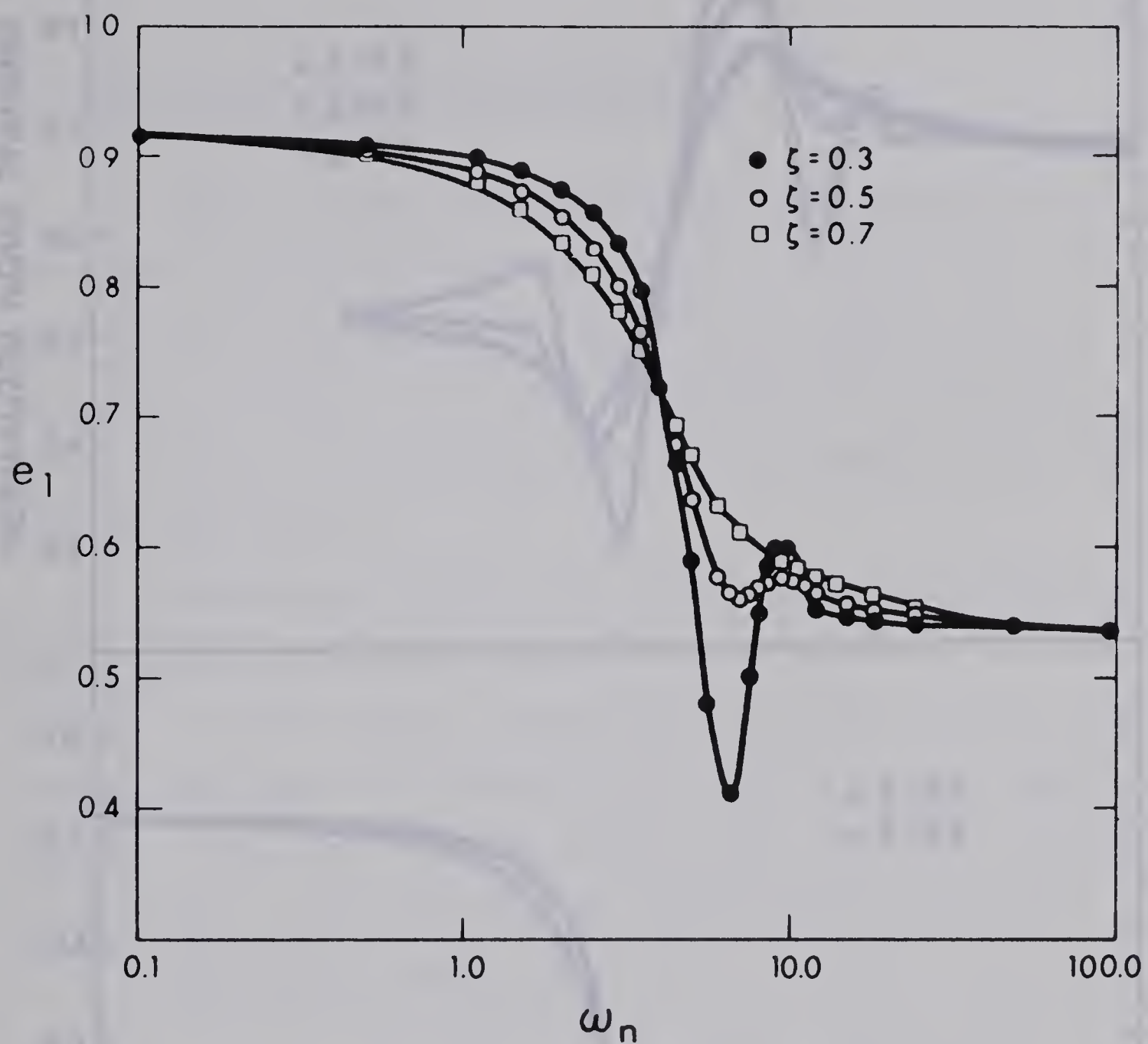


FIGURE 5.14 PREDICTED e_1 FOR PLCs OF $G_4(s)$ WITH $T=1$ SEC.

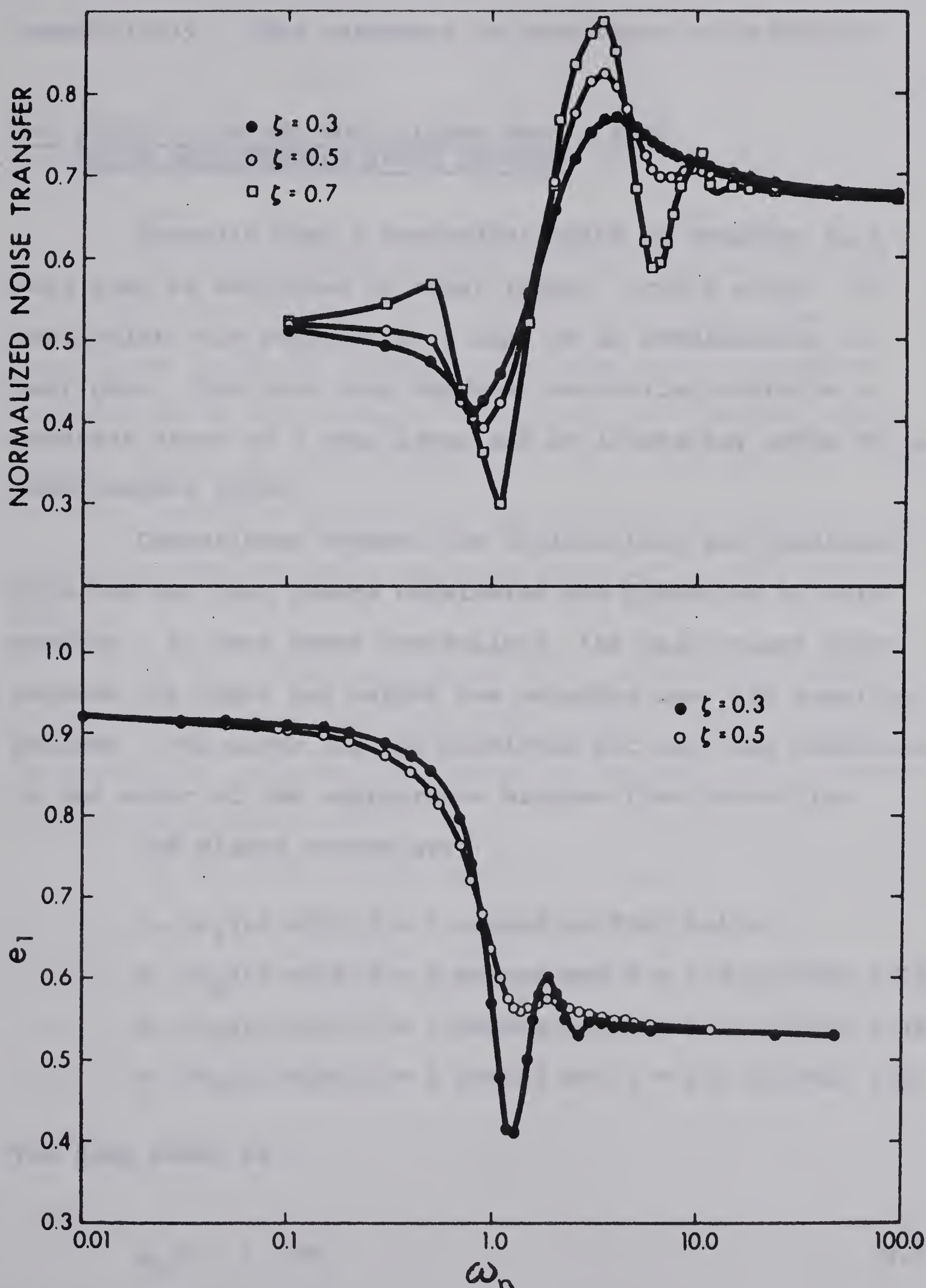


FIGURE 5.15 NORMALIZED NOISE TRANSFERS USING PREDICTED PLCS OF $G_4(s)$ WITH $T=1$ SEC. USING GAUSS2

FIGURE 5.16 PREDICTED e_1 FOR PLCS OF $G_4(s)$ WITH $T=5$ SEC.

normalized noise transfer differences are -1.2% and -0.2% respectively. This agreement is considered satisfactory.

5-6 RESULTS FOR THE FOUR PLANTS WHEN A RAMP OR AN ACCELERATION INPUT IS USED

Normally when a controller which is deadbeat to a unit step is subjected to other inputs, errors occur. In particular, the response to a ramp or an acceleration is very poor. The unit step deadbeat controller exhibits a constant error to a ramp input and an increasing error to an acceleration input.

Comparisons between the minimum time and predicted PLCs for the four plants considered are presented in this section. To test these controllers, the mean square error between the input and output was measured over 128 sampling periods. The error for the predicted PLC was then normalized to the error of the appropriate minimum time controller.

The plants tested are

1. $G_1(s)$ with $T = 1$ second in FIG. 5.17.
2. $G_2(s)$ with $T = 1$ second and $\beta = 1.0$ in FIG. 5.18.
3. $G_3(s)$ with $T = 1$ second and $\beta = 5.0$ in FIG. 5.19.
4. $G_4(s)$ with $T = 1$ second and $\zeta = 0.5$ in FIG. 5.20.

The ramp input is

$$u_r(t) = NT \quad (5.24)$$

and the acceleration input is

$$u_a(t) = (NT)^2 \quad (5.25)$$

where N is the number of the sampling period and $0 \leq N \leq 128$.

The graphs shown in FIG. 5.17 to 5.19 are plotted over very small ranges of α . For $G_1(s)$ to $G_3(s)$, the improvement of the PLCs over the minimum time controller occurs only in this region. For values of α outside these ranges, the error of the PLC was larger. In particular for $\alpha > 1.0$, the PLC responds more slowly than the minimum time controller and therefore the normalized error rises rapidly with α .

In FIG. 5.20, the values of ω_n plotted are limited as the normalized errors are greater than 1.0 for ω_n outside these values. This is similar to the results for the other plants considered.

For the four plants tested, the maximum improvement in the error is significant. The maximum improvement is slightly greater for the acceleration error. The range of plant parameters for which this improvement is obtained are very limited in each case.

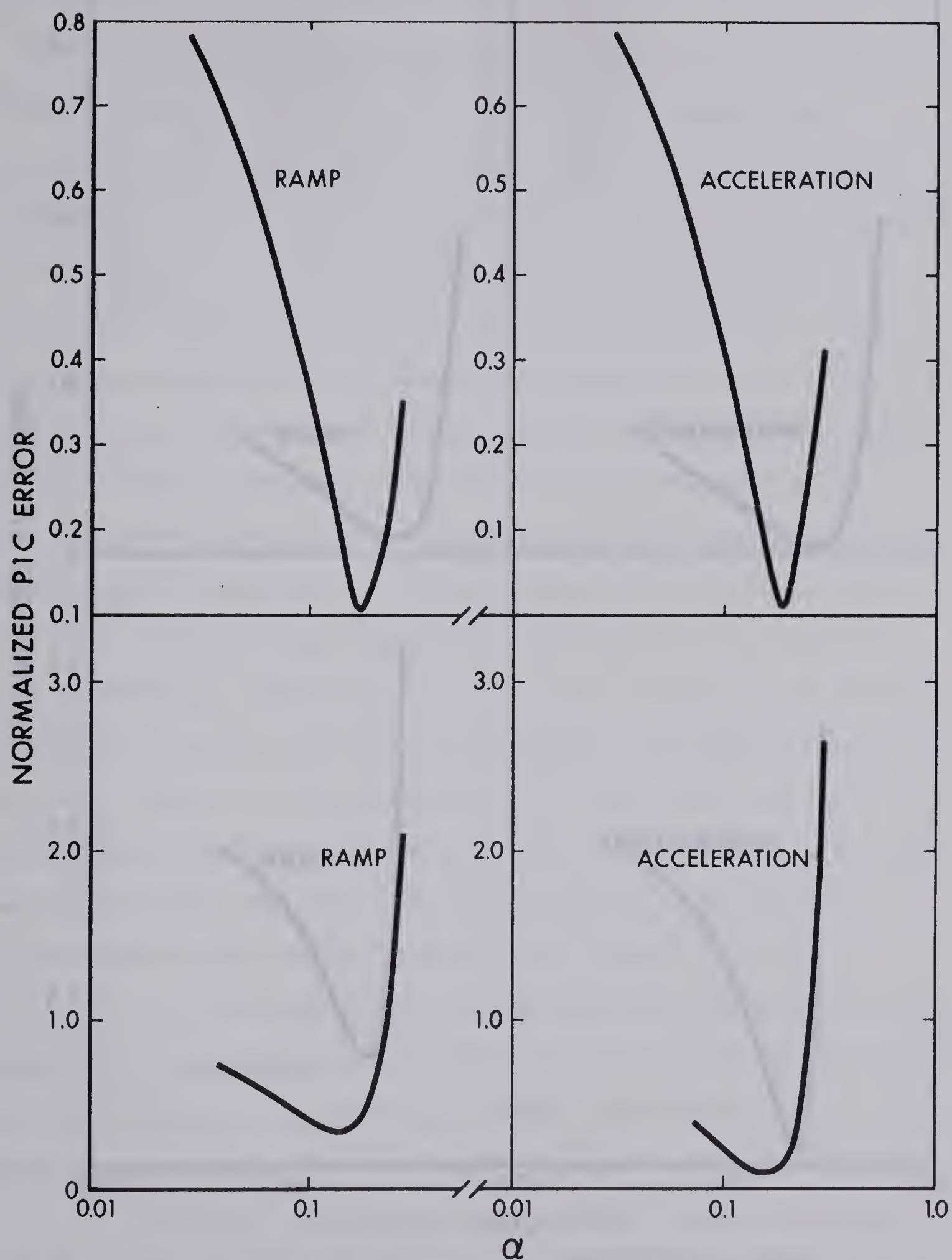


FIGURE 5.17 NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED PLCs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 5.18 NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED PLCs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.

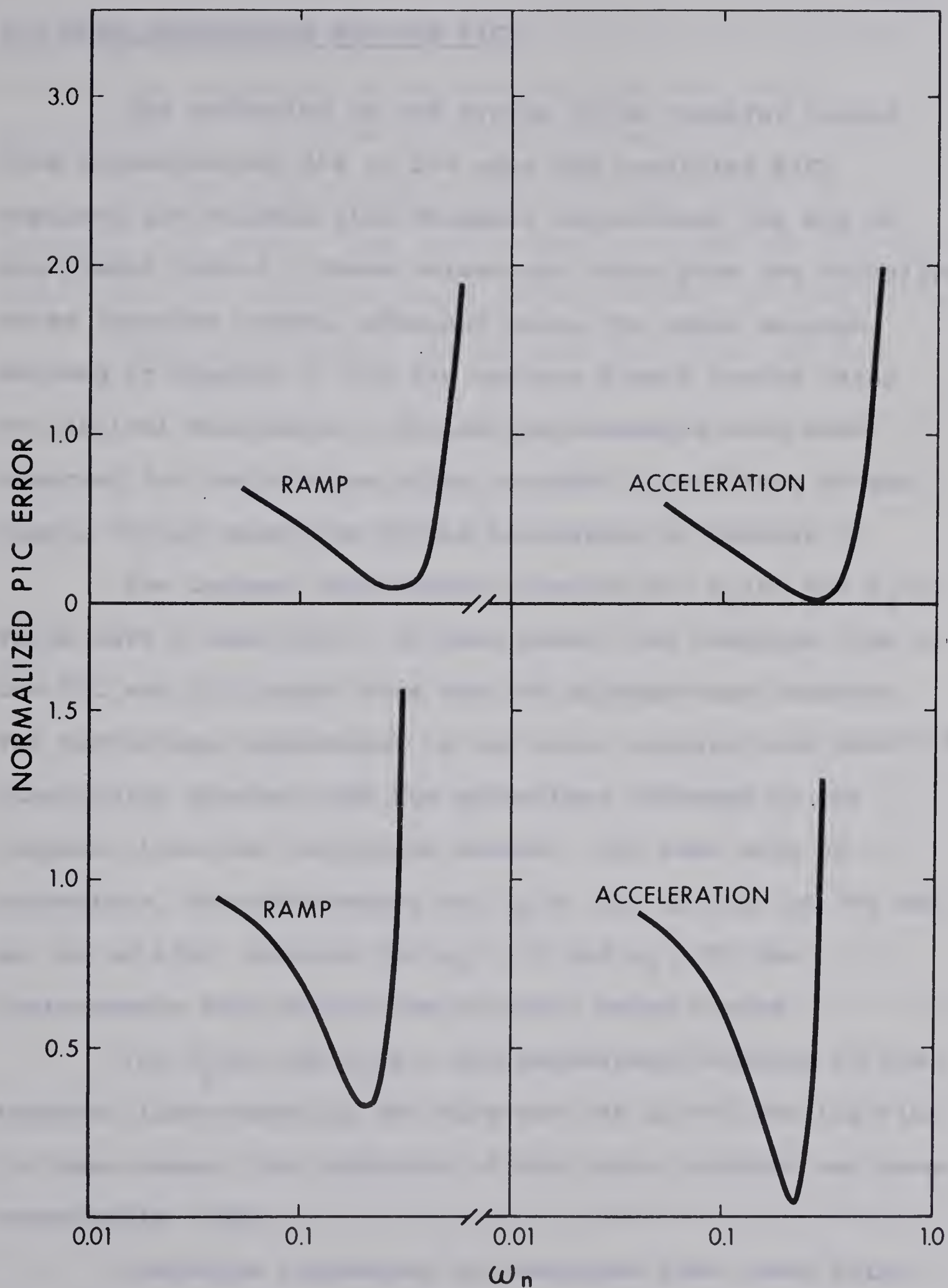


FIGURE 5.19 NORAMLIZED RAMP AND ACCELERATION ERRORS USING PREDICTED P1Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.

FIGURE 5.20 NORMALIZED RAMP AND ACCELERATION ERRORS USING PREDICTED P1Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC.

5-7 SOME CONCLUSIONS FOR THE PLCs

The reduction of the system noise transfer varied from approximately 25% to 50% when the predicted PLCs replaced the minimum time deadbeat controllers for any of the plants tested. These values are taken from the normalized noise transfer curves, obtained using the noise sources defined in Chapter 4, for the various plants tested using the digital simulation. Similar improvements were also observed for the minimum noise transfer controllers of the plants tested using the hybrid simulation in Chapter 3.

The largest improvements occurred for $G_2(s)$ and $G_4(s)$ which have 4 term PLCs. In both cases, the response time of the PLC was 1/3 longer than for the minimum time deadbeat. The percentage improvement in the noise transfer was generally numerically greater than the percentage increase in the response time for the plants tested. For some sets of parameters, the improvement for $G_4(s)$ was as high as 70% and as low as 15%. However for $\omega_n < .5$ and $\omega_n > 5$, the improvements fell within the original range stated.

For $G_1(s)$ and $G_3(s)$, the percentage increase in the response time caused by the PLCs was 50% as $n=3$ for the PLCs. In these cases, the reduction of the noise transfer was never numerically $> 50\%$.

Therefore increasing the response time, when using the predicted PLCs, gave the best noise transfer improvement for higher order systems. However, the improvement for all systems tested was substantial when the predicted PLCs were

used. Therefore the trade-off of increased response time and greater controller complexity for improved noise performance is justified for the predicted PLCs.

It was stated earlier that the input noise is assumed to be white. The grid searches carried out for each plant used the non-white noise sources defined in Chapter 4.

From these tests, two conclusions may be drawn.

1. For all plants tested, the agreement between the predicted and "grid searched" PLCs was generally very close. The noise transfer differences were always $< 3.7\%$ and generally were $< 2.0\%$. The e_1 differences were always $< 7.0\%$ for any noise source used. This indicates that the predicted PLCs are at least a good approximation for a variety of non-white noise sources. The results from the hybrid simulation also support this statement.
2. The agreement between the predicted and "grid searched" PLCs was best for both e_1 and the noise transfers when the GAUSS generated noise sources were used. These have lower deviations from a uniform frequency distribution between 0 and $\omega_s/2$ than do the RANDU generated noise sources. This supports the assumption that the NCE criterion is derived for a white noise input as the GAUSS noise sources are "whiter" in terms of

their frequency distribution than are the RANDU noise sources.

For the plants $G_1(s)$, $G_3(s)$ and $G_4(s)$, define the respective dimensionless constants αT , αT and $\zeta \omega_n T$ as τ . For $G_2(s)$, define the two dimensionless constants αT and βT as τ_1 and τ_2 . These constants are analogous in form to time constants.

In this chapter, all noise transfer curves were plotted for a sampling rate of 1 second. All these curves had one common tendency. When τ was very large or very small, the rate of change of the noise transfer curve for changes in τ was very slow. Between these two regions there was a transition zone. The transition zone extended from $\tau \approx 0.1T$ to $\tau \approx 10T$ in all cases.

One reason for this transition zone is the interaction of the values in τ . For $\tau < 0.1T$, the dominant terms are α , $\omega_n \zeta$ or β depending on the plant under consideration. For $\tau > 10T$, T is the dominant term. In the transition zone, there is no clearly dominant term.

Two constants τ_1 and τ_2 exist for $G_2(s)$. Either one may cause a transition zone to exist if the other is held constant.

Within this transition zone, the absolute maximum and the absolute minimum noise reductions occurred for all plants tested. For $G_1(s)$, the maximum reduction occurred at $\tau = 0.7$ and the minimum reduction occurred at $\tau \approx 3.5$. The appropriate

values of τ may also be found for the other plants tested.

Therefore a set of plant parameters exists which give the best possible noise improvement for a specific plant type. If the plant parameters or the system sampling rate may be altered to any extent, the constant τ can be adjusted to give the best noise reduction.

The sensitivity of a deadbeat controller is effectively infinity as the response to a unit step is either deadbeat or it is not. However if a small parameter variation occurred in the plant, the difference between the ideal deadbeat response and the perturbed response was generally not very large. Both the minimum time deadbeat controllers and the predicted PlCs were tested for this error.

For the four plants, one of the parameters was varied by $\pm 5\%$ and the resultant errors in the response to a unit step were evaluated over 128 sampling periods. For $G_1(s)$, $G_2(s)$ and $G_3(s)$, this parameter was α . For $G_4(s)$, ω_n was varied. In all cases, the differences between the errors measured for the minimum time controllers and the predicted PlCs were less than 2%. Depending on the parameters, the error could be lower for either type of controller. Therefore the predicted PlCs caused no degradation in this respect when compared to the corresponding minimum time deadbeat controller.

CHAPTER VI

CONTROLLERS EXTENDED TWO TERMS

In this chapter, the results of the purely digital simulation of the controllers extended by two terms are presented. To simplify the presentation, a controller extended by two terms is referred to as a P2C. This abbreviation stands for Plus Two Terms Controller.

The algorithm used to calculate the P2Cs is derived in the first section. This derivation is based on the design criterion NCE. In the following sections, the results obtained from the digital simulation of the four plants listed in Chapter 1 are presented. These results are presented in the same format used in Chapter 5. To compare the PlCs and the P2Cs, a graph showing the normalized noise transfers obtained using both types of predicted controllers is given for each plant tested. The final section of the chapter presents some conclusions derived from the test results.

6-1 ALGORITHM USED IN THE DIGITAL SIMULATION TO CALCULATE THE PREDICTED P2Cs

In this section, the algorithm used to calculate the predicted P2Cs is derived from NCE. It was shown in Section 5-2, that one extra term gave one additional degree of freedom in the calculation of the $D(z)$. This degree of freedom was represented by the choice of e_1 .

Similarly adding two extra terms to the controller corresponds to two extra degrees of freedom in the calculation. The choices of e_1 and e_2 represent these two degrees of freedom.

The controllers are calculated by the method derived in Section 2-4. Expressing the remaining coefficients of the P2C as functions of e_1 and e_2 yields:

$$h_0 = a_1 e_1 + a_2 e_2 + a_3 \quad (6.1)$$

$$h_1 = a_4 e_1 + a_5 e_2 + a_6 \quad (6.2)$$

$$h_2 = a_7 e_1 + a_8 e_2 + a_9 \quad (6.3)$$

$$h_3 = a_{10} e_1 + a_{11} e_2 + a_{12} \quad (6.4)$$

$$h_4 = a_{13} e_1 + a_{14} e_2 + a_{15} \quad (6.5)$$

$$e_3 = b_1 e_1 + b_2 e_2 + b_3 \quad (6.6)$$

$$e_4 = b_4 e_1 + b_5 e_2 + b_6 \quad (6.7)$$

where the order of the P2C is $n \leq 4$. These coefficients are given in Table 6.1 for the four plants tested. The coefficients are defined in terms of the constants derived in Chapter 1 for each plant.

For controllers with $n \leq 4$, the form of the criterion NCE is

$$\begin{aligned}
\text{NCE} = & h_0^2 + (h_1 - e_1 h_0)^2 + (h_2 - e_1 h_1 - e_2 h_0 + \\
& e_1^2 h_0)^2 + (h_3 - e_1 h_2 - e_2 h_1 - e_3 h_0 + \\
& e_1^2 h_1 - e_1^3 h_0 + 2e_1 e_2 h_0)^2 + (h_4 - e_1 h_3 - \\
& e_2 h_2 - e_3 h_1 - e_4 h_0 + e_1^2 h_2 + 2e_1 e_2 h_1 + \\
& 2e_1 e_3 h_0 + e_2^2 h_0 - 3e_1^2 e_2 h_0 - e_1^3 h_1 + e_1^4 h_0)^2
\end{aligned}
\tag{6.8}$$

Since the coefficients defined in (6.1) to (6.7) are functions of e_1 and e_2 , NCE may be written in the same form. Define NCE_2 as the criterion for the system with two extra sampling periods. Rewriting NCE in this form now yields:

$$\text{NCE}_2 = f(e_1, e_2) = \sum_{i=0}^n \text{NCE}_{2i} \tag{6.9}$$

where

$$\begin{aligned}
\text{NCE}_{20} &= h_0^2 \\
&\vdots
\end{aligned}
\tag{6.10}$$

$$\begin{aligned}
\text{NCE}_{24} = & (h_4 - e_1 h_3 - e_2 h_2 - e_3 h_1 - e_4 h_0 + \\
& e_1^2 h_2 + 2e_1 e_2 h_1 + 2e_1 e_3 h_0 + e_2^2 h_0 - \\
& 3e_1^2 e_2 h_0 - e_1^3 h_1 + e_1^4 h_0)^2
\end{aligned}
\tag{6.11}$$

The substitution of (6.1) to (6.7) into (6.8) yields NCE_2 which contains products of the powers of e_1 and e_2 . Thus it is not possible to find the values of e_1 and e_2 which correspond to the minimum of NCE_2 by simple differentiation as in Chapter 5.

An alternate method using the partial differentiation of NCE_2 with respect to e_2 for fixed values of e_1 is derived in the remainder of this section.

Substituting (6.1) to (6.7) into (6.8) yields in simplified form:

$$NCE_{20} = a_2^2 e_2^2 + 2a_2 b_{01} e_2 + b_{01}^2 \quad (6.12)$$

$$NCE_{21} = b_{11}^2 e_2^2 + 2b_{11} b_{12} e_2 + b_{12}^2 \quad (6.13)$$

$$NCE_{22} = a_2^2 e_2^4 - 2a_2 b_{21} e_2^3 + (b_{21}^2 - 2a_2 b_{22}) e_2^2 + 2b_{21} b_{22} e_2 + b_{22}^2 \quad (6.14)$$

$$NCE_{23} = b_{31}^2 e_2^4 + 2b_{31} b_{32} e_2^3 + (b_{32}^2 + 2b_{31} b_{33}) e_2^2 + 2b_{32} b_{33} e_2 + b_{33}^2 \quad (6.15)$$

$$NCE_{24} = b_{41}^2 e_2^4 + 2b_{41} b_{42} e_2^3 + (b_{42}^2 + 2b_{41} b_{43}) e_2^2 + 2b_{42} b_{43} e_2 + b_{43}^2 \quad (6.16)$$

where

$$b_{01} = a_1 e_1 + a_3$$

$$b_{11} = a_5 - a_2 e_1$$

$$b_{12} = -a_1 e_1^2 + (a_4 - a_3) e_1 + a_6$$

$$b_{21} = (a_8 - a_9) - (a_1 + a_5) e_1 + a_2 e_1^2$$

$$b_{22} = a_9 + (a_7 - a_6) e_1 + (a_3 - a_4) e_1^2 + a_1 e_1^3$$

$$b_{31} = 2a_1 e_1 - a_2 b_2 - a_5$$

$$b_{32} = -a_2 e_1^3 + (2a_1 + a_5) e_1^2 + (-a_4 - 2a_1 b_2 +$$

$$2a_3 - a_8) e_1 + (a_{11} - a_6 - 2a_2 b_3)$$

$$b_{33} = -a_1 e_1^4 + (a_4 - a_3) e_1^3 + (a_6 - a_7 - a_1 b_1) e_1^2 +$$

$$(a_{10} - a_9 - 2a_1 b_3) e_1 + (a_{12} - a_3 b_3)$$

$$b_{41} = -3a_2 e_1^2 + (a_1 + 2a_2 b_2 + 2a_5) e_1 + (-a_8 - a_2 b_5 -$$

$$a_5 b_2 + a_3)$$

$$b_{42} = a_2 e_1^4 - (a_5 + 3a_1) e_1^3 + (4a_1 b_2 - 3a_3 +$$

$$2a_4 + a_8) e_1^2 + (-a_{11} - a_7 - 2a_4 b_2 - 2a_1 b_5 +$$

$$+2a_6 + 4a_2 b_3) e_1 + (a_{14} - a_9 - 2a_5 b_3 - 2a_2 b_6)$$

$$\begin{aligned}
b_{43} = & a_1 e_1^5 + (a_3 - a_4) e_1^4 + (2a_1 b_1 + a_7 - a_6) e_1^3 + \\
& (4a_1 b_3 + a_9 - a_1 b_4 - a_{10} - a_4 b_1) e_1^2 + \\
& (a_{13} - a_{12} - 2a_4 b_3 - 2a_1 b_6 + 2a_3 b_3) e_1 + \\
& (a_{15} - a_6 b_3 - a_3 b_6)
\end{aligned} \tag{6.17}$$

Collecting (6.12) to (6.16) yields the simplified expression:

$$\begin{aligned}
NCE_2 = \sum_{i=0}^n NCE_{2i} = & D_4 e_2^4 + D_3 e_2^3 + D_2 e_2^2 + \\
& D_1 e_2 + D_0
\end{aligned} \tag{6.18}$$

where $2 \leq n \leq 4$ and

$$D_i = f_i(e_1) ; \quad i = 0, 1, \dots, 4 \tag{6.19}$$

For $0 \leq n \leq 1$, NCE_2 cannot be calculated as no P2C can exist for these values.

When D_0 to D_4 are evaluated, then (6.18) is expressed only as a function of e_2 . If the values of e_1 are known, then D_0 to D_4 may be calculated. One way of obtaining these values of e_1 is by iteration over $0 \leq e_1 \leq 1.0$ which is required for deadbeat.

For a particular value of e_1 , equation (6.18) may be rewritten in simplified form as:

$$NCE_2 = [f(e_2)] \Big|_{e_1} \quad (6.20)$$

Setting the partial derivative of (6.20) with respect to e_2 to zero yields:

$$\frac{\partial}{\partial e_2} [NCE_2] = \frac{\partial}{\partial e_2} \left[[f(e_2)] \Big|_{e_1} \right] = 0 \quad (6.21)$$

Substituting (6.18) into (6.21) yields:

$$4(D_4 \Big|_{e_1})e_2^3 + 3(D_3 \Big|_{e_1})e_2^2 + 2(D_2 \Big|_{e_1})e_2 + (D_1 \Big|_{e_1}) = 0 \quad (6.22)$$

From (6.22), solving for the value of e_2 which satisfies the deadbeat requirements:

$$0 \leq e_2 \leq 1.0 \quad (6.23a)$$

$$e_2 \leq e_1 \quad (6.23b)$$

yields the value of e_2 which corresponds to the minimum value of NCE_2 for a particular value of e_1 . The criterion NCE_2 may be evaluated by substituting the value of e_1 and the corresponding value of e_2 into (6.18) and (6.19).

From the values of NCE_2 calculated for all e_1 values

	$\frac{G_1(s)}{G_2(s)}$	
a ₁	-1/R1	-1.0/R7
a ₂	0	0
a ₃	-a ₁	-a ₁
a ₄	a ₃ (A + 1 + R2 a ₃)	a ₃ (A + B) + a ₁ (1 + a ₃ R8)
a ₅	a ₁	a ₁
a ₆	-a ₄ + a ₃	-a ₃ ² R8 + a ₁ (A + B)
a ₇	a ₁ (A(1+R2 a ₃) + R2(1+a ₃ R4)/(R1+R2))	a ₁ AB + (A+B)(1+a ₃ R8) a ₁
a ₈	a ₃ (1 + A)	a ₃ (A+B) + (1+a ₃ (U+R8))/(U+R7+R8)
a ₉	a ₃ ² R2(A+R2)/(R1+R2))	a ₃ AB+a ₃ ² R8(A+B) + a ₃ (a ₃ R8(U+R8)-U)/(U+R7+R8)
a ₁₀	a ₃ AR2(1+a ₃ R2)/(R1+R2)	a ₃ (1+a ₃ R8)AB + R8(A+B)(a ₃ ² U+a ₃ (1+a ₃ R8))/(U+R7+R8)
a ₁₁	a ₁ A	a ₁ AB-(A+B)(1+a ₃ (R8+U))/(U + R7 + R8)
a ₁₂	(a ₃ a ₁ ² R2 ²)/(R1 + R2)	-a ₃ ² AB(R8)+(A+B)a ₁ (a ₃ R8(U+R8)-U)/(U+R7 + R8)
a ₁₃	0	a ₁ AB(a ₃ U + 1 + a ₃ R8) R8/(U + R7 + R8)
a ₁₄	0	AB(1 + a ₃ (R8 + U))/(U + R7 + R8)
a ₁₅	0	a ₃ AB(a ₃ R8(R8 + U) - U)/(U + R7 + R8)
b ₁	-(a ₁₀ R2)/A	(a ₁₃ U)/(AB)+a ₃ U(1+a ₃ R8)+a ₁ R8 ² (a ₃ U+1+a ₃ R8)/(U+R7+R8)
b ₂	a ₃ R2	(a ₁₄ U)/(AB)+a ₁ U+R8(1+a ₃ (U+R8))/(U + R7 + R8)
b ₃	(b ₂ ² R2)/(R1 + R2)	(a ₁₅ U)/(AB)-a ₃ ² (R8)U+a ₃ R8(a ₃ R8(R8+U)-U)/(U+R7+R8)
b ₄	0	(a ₁₃ U)/(AB)
b ₅	0	(a ₁₄ U)/(AB)
b ₆	0	(a ₁₅ U)/(AB)
ORDER OF P2C	n = 4	n = 5

TABLE 6.1(a) COEFFICIENTS OF THE P2C FOR CALCULATION AS FUNCTIONS OF e₁ AND e₂ FOR G₁(s) AND G₂(s)

	$\frac{G_3(s)}{G_4(s)}$	
a_1	$-1/B$	$-1/A1$
a_2	0	0
a_3	$-a_1$	$-a_1$
a_4	$a_3A + a_1(a_1C - 1)$	$a_1AC8 + a_3(1 + a_3A2)$
a_5	$a_1^2A - a_1^2C$	$a_1^2AC8 - a_3^2A2$
a_6	$a_3A(a_1C - 1) + a_3C(a_1C - 1)/(C+B)$	$a_1AC9+a_3AC8(1+a_3A2)+(a_3A3+(A2+A3))(1+a_3A2)a_1/(A1+A2+A3)$
a_7	$a_3A + (1 + a_3C)/(C + B)$	$a_1AC8 + (1 + a_3(A2 + A3))/(A1 + A2 + A3)$
a_8	$a_3^2AC + a_3^2C^2/(C + B)$	$a_3AC9-a_3^2(A2)AC8+(a_1A3+a_3^2A2(A2+A3))/(A1 + A2 + A3)$
a_9	$a_1AC(a_1C - 1)/(C + B)$	$a_3AC9(1+a_3A2)+AC8(a_3A3-(A2+A3)(1+a_3A2)a_3)/(A1+A2+A3)$
a_{10}	$-A(1 + a_3C)/(C + B)$	$a_1AC9 + AC8(1+a_3(A2+A3))/(A1 + A2 + A3)$
a_{11}	$-(a_3^2AC^2)/(C + B)$	$-a_3^2(A2)AC9 + AC8(a_1A3+a_3^2A2(A2 + A3))/(A1+A2+A3)$
a_{12}	0	$AC9(a_3A3 + (A2+A3)(1+a_3A2)a_1)/(A1 + A2 + A3)$
a_{13}	0	$AC9(1 + a_3(A2 + A3))/(A1 + A2 + A3)$
a_{14}	0	$AC9(a_1A3 + (A2 + A3)a_3^2A2)/(A1 + A2 + A3)$
a_{15}	0	0
b_1	$-(a_{10}C)/A$	$((A2+A3)(a_3A3+(A2+A3)(1+a_3A2)a_1)/(A1+A2+A3))+a_3A3(1 + a_3A2)$
b_2	$-(a_{11}C)/A$	$a_1A3 + (A2+A3)(1 + a_3(A2+A3))/(A1 + A2 + A3)$
b_3	$-(a_{12}C)/A$	$-a_3^2(A2)A3+(A2+A3)(a_1A3+a_3^2A2(A2+A3))/(A1+A2+A3)$
b_4	0	$(a_{13}A3)/AC9$
b_5	0	$(a_{14}A3)/AC9$
b_6	0	$(a_{15}A3)/AC9$

TABLE 6.1(b) COEFFICIENTS OF THE P2C FOR CALCULATION AS FUNCTIONS OF e_1 AND e_2 FOR $G_3(s)$ AND $G_4(s)$

obtained by iteration, the global minimum value of NCE_2 may be selected. The values of e_1 and e_2 which correspond to this value of NCE_2 are then the predicted values for the plant under consideration.

The method of calculation may be outlined as follows.

1. Write NCE_2 entirely as a polynomial in e_2 where the coefficients are functions of e_1 .
2. Iterate e_1 over the range 0 to 1.0 to calculate these coefficients.
3. For each e_1 value, take the partial derivative of NCE_2 with respect to e_2 and set it to zero. Solve for the real root of e_2 which satisfies (6.23).
4. Evaluate NCE_2 at each value of e_1 and the corresponding value of e_2 . The value of NCE_2 which is the global minimum yields the e_1 and e_2 values of the predicted P2C.

6-2 RESULTS FOR P2Cs OF $G_1(s) = 1/s(s + \alpha)$

This plant was tested with its P2Cs using the same test procedure as used for the PlCs in Chapter 5.

The sampling rate was set at $T = 1$ second and α was varied from 0.125 to 32.0. The predicted values of e_1 and e_2 are shown in FIG. 6.1 plotted versus α .

FIG. 6.2 is a plot of the normalized noise transfers of the systems containing the predicted P2Cs versus α . To

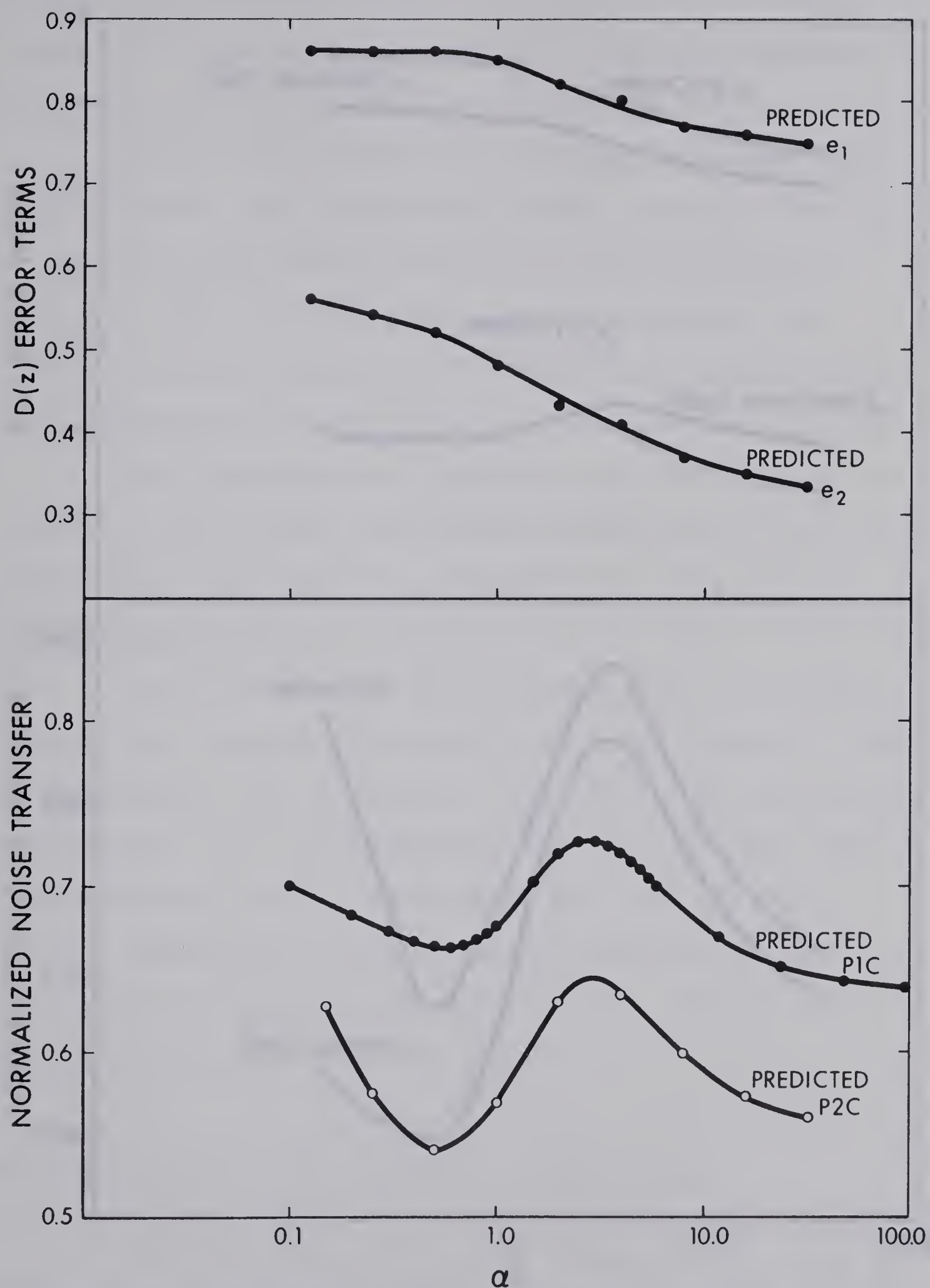


FIGURE 6.1 PREDICTED e_1 AND e_2 FOR P2Cs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 6.2 NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2

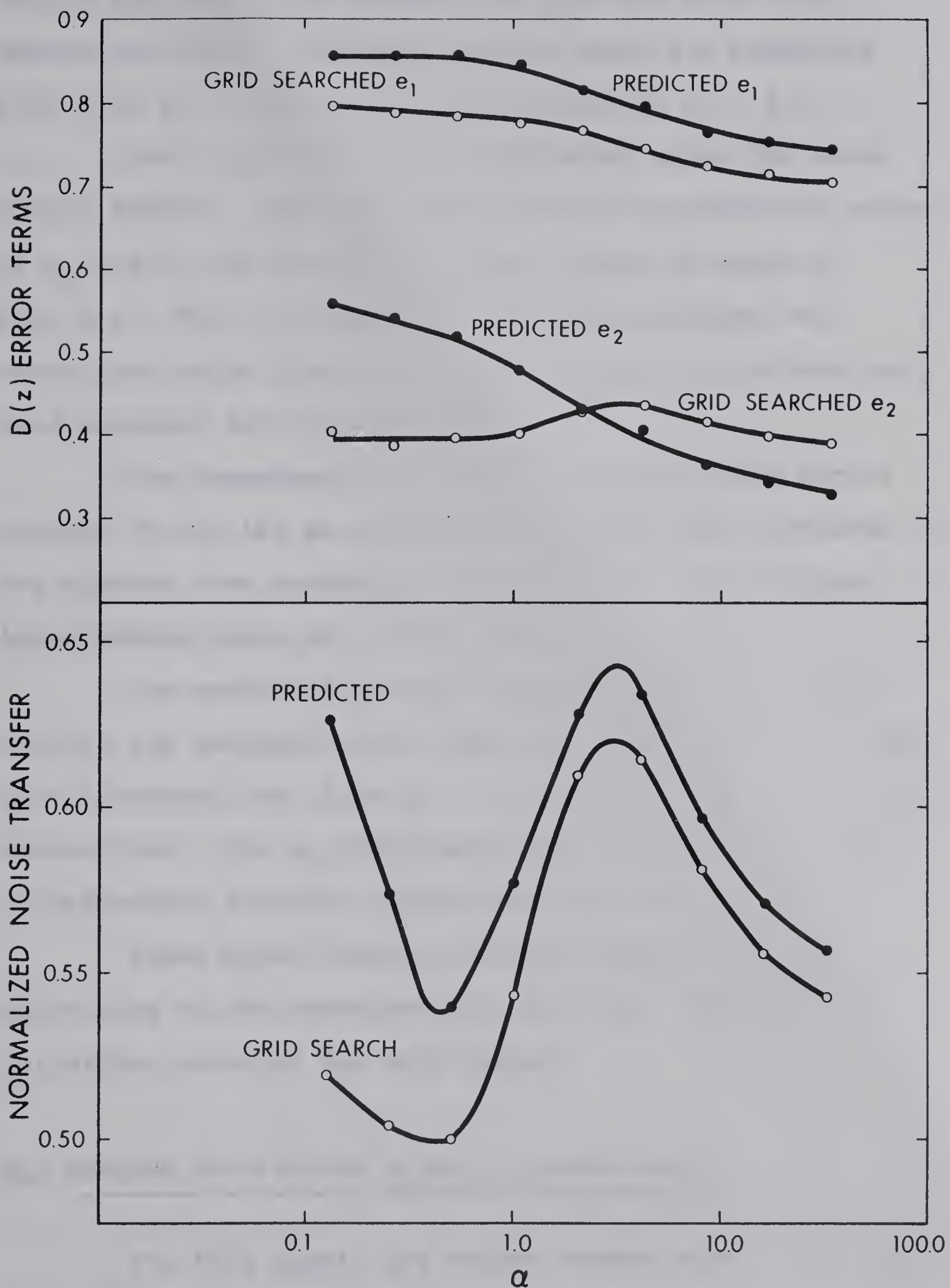


FIGURE 6.3 PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 6.4 NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2

obtain the comparison between the PLCs and P2Cs, the normalized noise transfers obtained using the predicted PLCs with $T = 1$ second are also plotted on FIG. 6.2.

Next a grid search was performed using the noise source GAUSS2. The comparison between the predicted values of e_1 and e_2 and the grid searched values is shown in FIG. 6.3. FIG. 6.4 shows the comparison between the normalized noise transfers for the systems containing the grid searched and predicted P2Cs.

The improvement of the P2Cs over the PLCs varies between 7% and 12% as calculated by GAUSS2 when referred to the minimum time deadbeat. From FIG. 6.2, the greatest improvements occur at $\alpha \approx 0.5$ and for $\alpha > 10.0$.

The predicted and grid searched results in this chapter are compared in the same way as in Chapter 5. The e_1 differences are -7.0% at $\alpha = 0.125$ and -5.3% at $\alpha = 32.0$. Respectively, the e_2 differences are -27% and +17%. The corresponding transfer differences are -20% and -5.3%.

These noise transfer results exhibit the same tendencies as the transfers for the PLCs. However, the variations involved are much larger.

6-3 RESULTS FOR P2Cs OF $G_2(s) = 1/s(s+\alpha)(s+\beta)$

For this plant, the values tested were $\beta = 1.0$ and $T = 1$ second with α varied from 0.2 to 32.0.

FIG. 6.5 gives the predicted values of e_1 and e_2 for

this plant. FIG. 6.6 shows the comparison between the normalized noise transfers obtained using the predicted PlCs and P2Cs for a GAUSS2 input. In FIG. 6.7, the grid search calculated values of e_1 and e_2 using GAUSS2 are compared to the predicted values. The comparison of the normalized noise transfers for the systems using the predicted and grid searched P2Cs is shown in FIG. 6.8.

A difficulty in calculating the predicted values of e_1 and e_2 for $\alpha \leq 1.0$ was encountered. The solution, as previously outlined, gave erroneous results for e_2 . To overcome this problem, the values of e_1 and e_2 were both iterated to find the minimum value of NCE_2 . This value then yielded the predicted e_1 and e_2 values plotted. For values of $\alpha > 1.0$, no difficulty was encountered. To check the results for $\alpha > 1.0$, the values of e_1 and e_2 were iterated and the global minimum value of NCE_2 calculated. For the values of $\alpha > 1.0$ checked, the results of both methods agreed.

The improvement in the normalized noise transfers when using the P2Cs in place of the PlCs is from 7% to 13% as calculated using GAUSS2. This is taken from FIG. 6.6.

The e_1 differences from FIG. 6.7 are -2.2% at $\alpha = 0.4$ and -8.3% at $\alpha = 32.0$. For e_2 , these differences are -7% and -15.2% respectively. The noise transfer differences are -6.8% and -3.6% respectively.

For this plant, the differences between the noise transfers are smaller than the error term differences. This

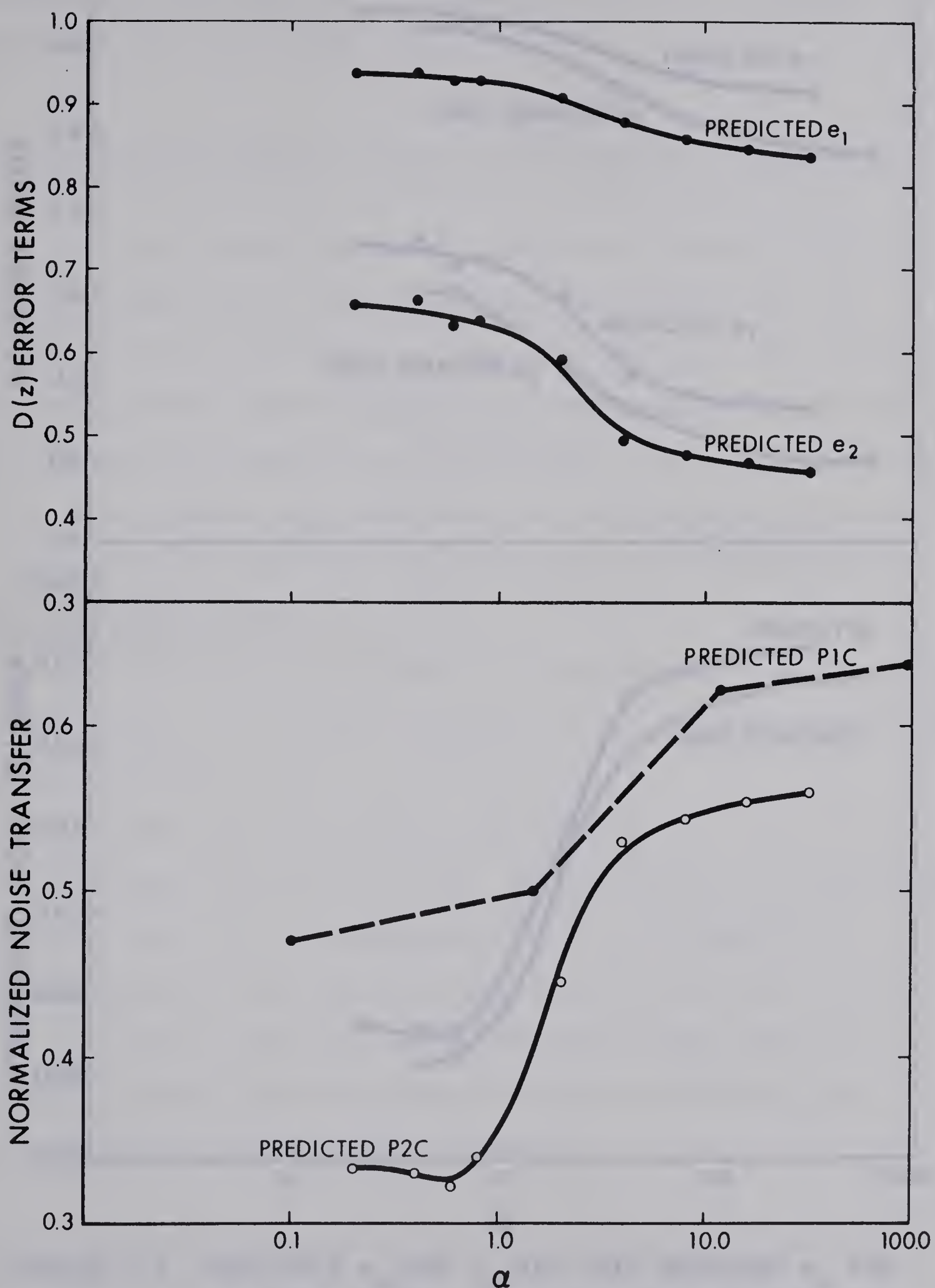


FIGURE 6.5 PREDICTED e_1 AND e_2 FOR P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.

FIGURE 6.6 NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. USING GAUSS2

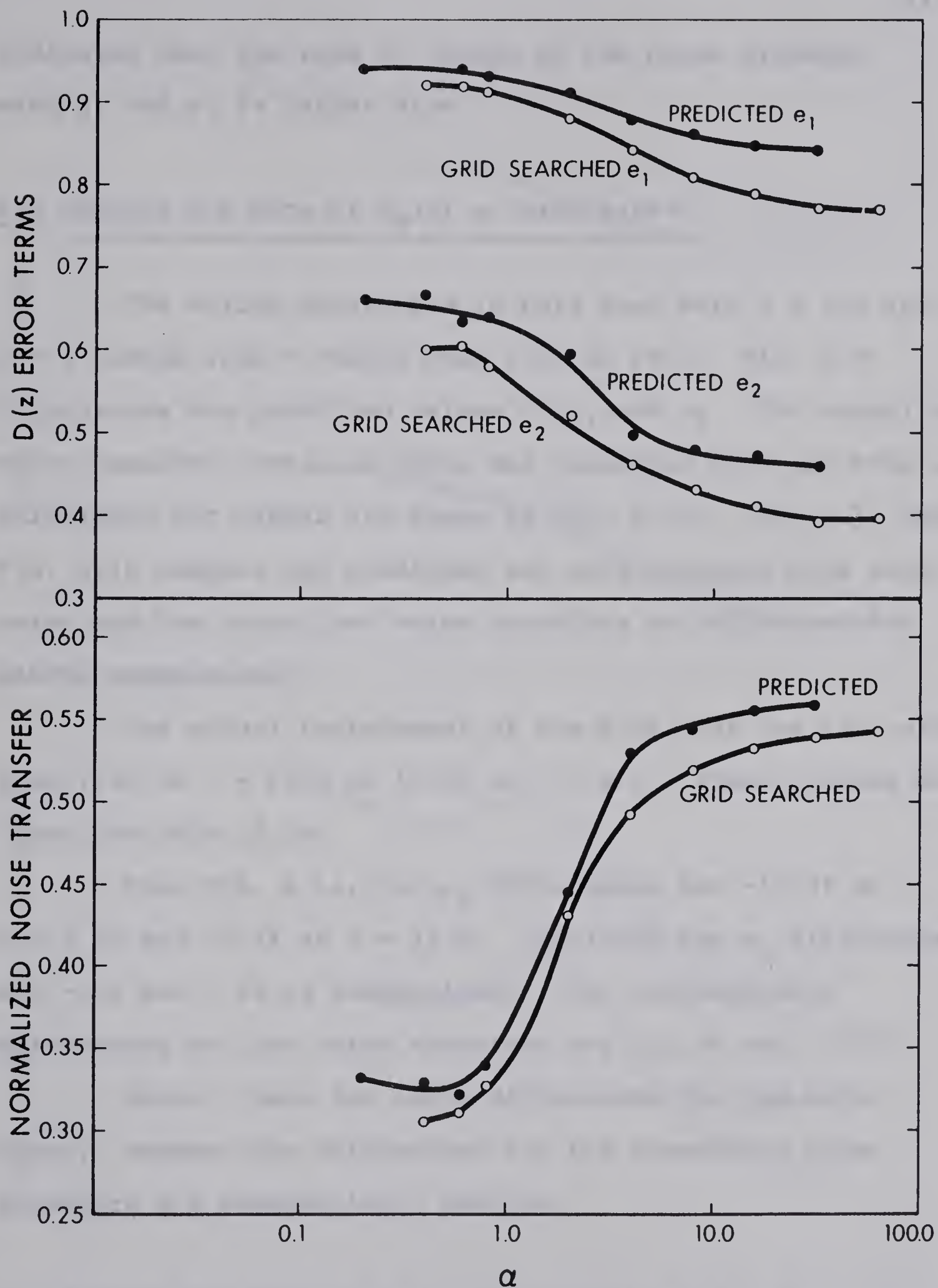


FIGURE 6.7 PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.

FIGURE 6.8 NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC. USING GAUSS2

indicates that the rate of change of the noise transfer with e_1 and e_2 is rather slow.

6-4 RESULTS FOR P2Cs of $G_3(s) = (s+\beta)/s(s+\alpha)$

The values considered in this case were $\beta = 5.0$ and $T = 1$ second with α varied from 0.25 to 64.0. FIG. 6.9 illustrates the predicted values of e_1 and e_2 . The normalized noise transfers obtained using the predicted PlCs and P2Cs as calculated for GAUSS2 are shown in FIG. 6.10. FIG. 6.11 and FIG. 6.12 compare the predicted and grid searched P2Cs error terms and the normalized noise transfers as calculated for GAUSS2 respectively.

The actual improvement of the P2Cs over the PlC varies from 7.0% at $\alpha = 64.0$ to 11.5% at $\alpha = 0.5$. These figures are taken from FIG. 6.10.

From FIG. 6.11, the e_1 differences are -13.4% at $\alpha = 0.25$ and -5.5% at $\alpha = 32.0$. Similarly the e_2 differences are -30% and + 28.1% respectively. The corresponding differences for the noise transfers are -10.0% and -3.2%.

Again, there are large differences for the error terms. However the differences for the normalized noise transfers are substantially smaller.

6-5 RESULTS FOR P2Cs OF $G_4(s) = 1/s(s^2 + 2\zeta\omega_n s + \omega_n^2)$

This plant was tested for $\zeta = 0.5$ and $T = 1$ second with ω_n varied from 0.25 to 64.0.

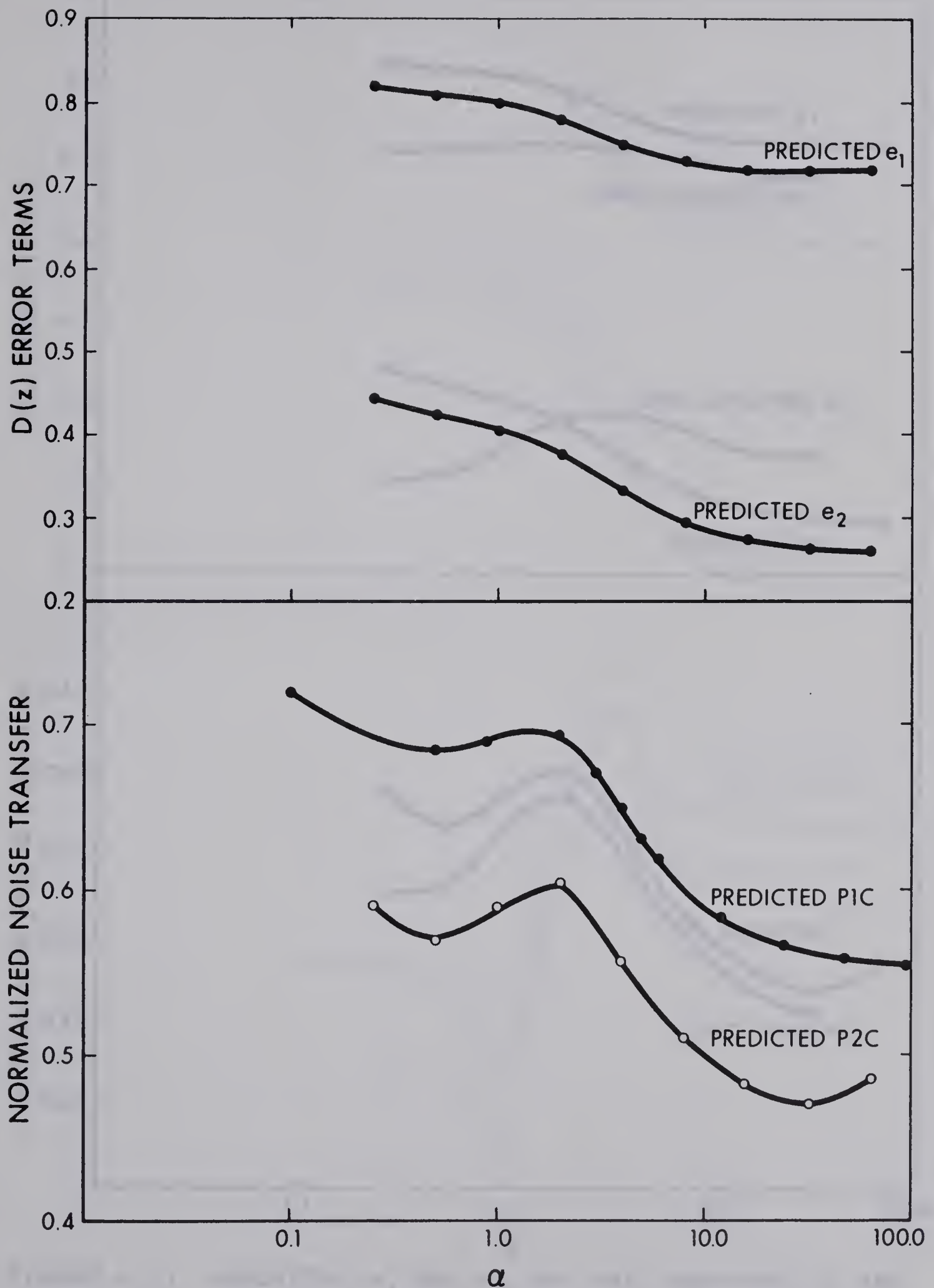


FIGURE 6.9 PREDICTED e_1 AND e_2 FOR P2Cs of $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.

FIGURE 6.10 NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.

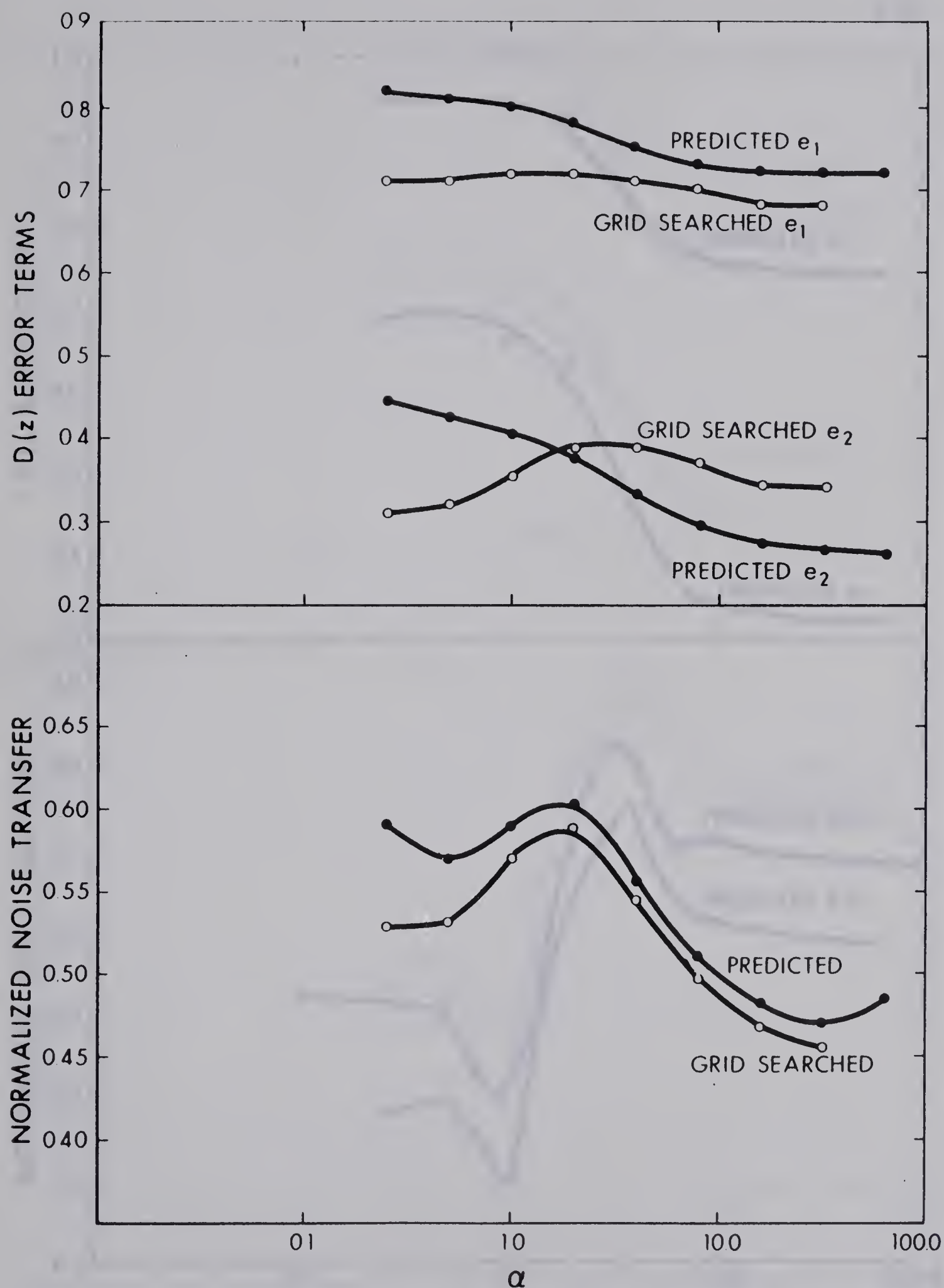


FIGURE 6.11 PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC.

FIGURE 6.12 NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_3(s)$ WITH $\beta=5$ AND $T=1$ SEC. USING GAUSS2

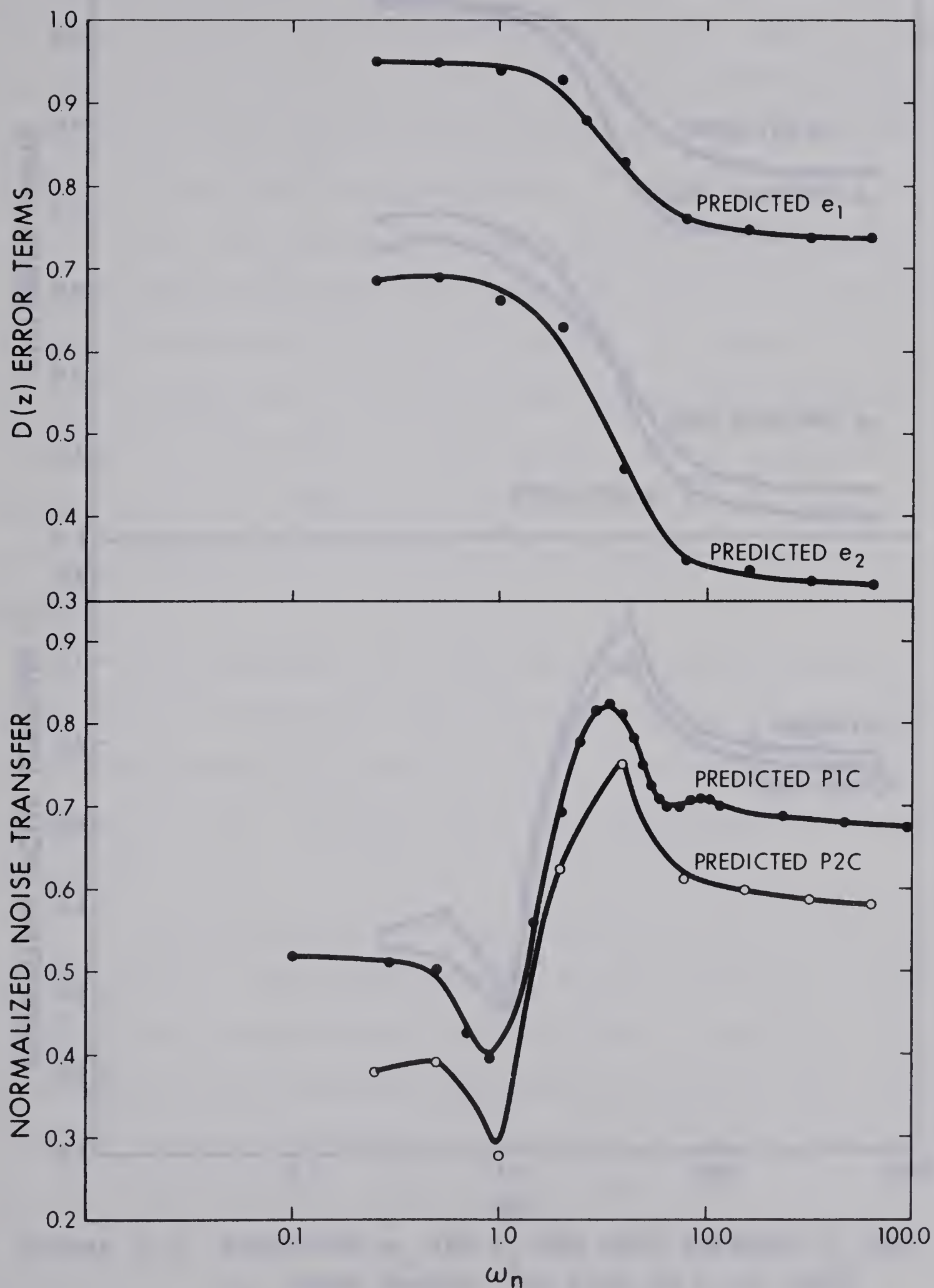


FIGURE 6.13 PREDICTED e_1 AND e_2 FOR P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC.

FIGURE 6.14 NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs AND P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC. USING GAUSS2

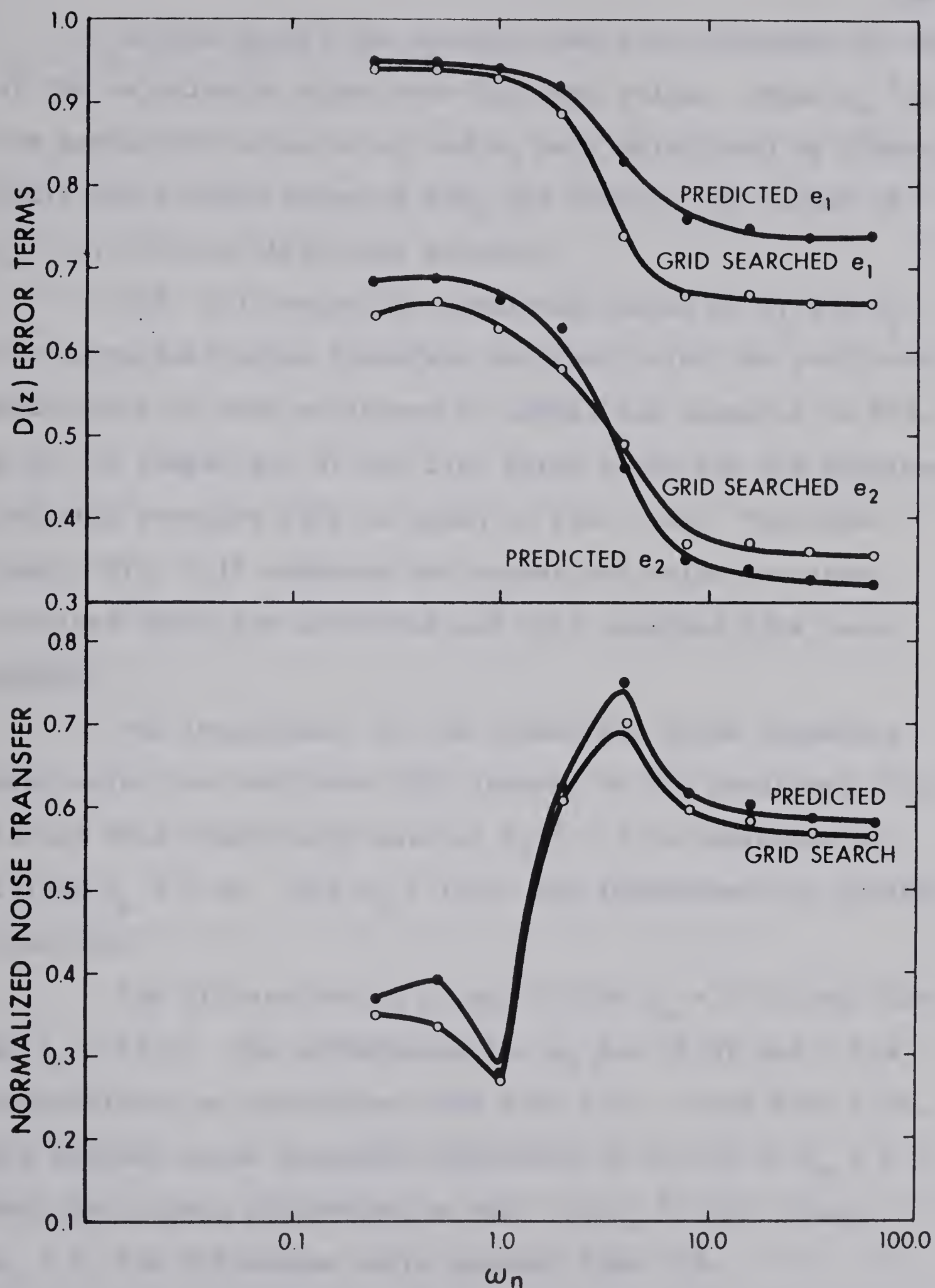


FIGURE 6.15 PREDICTED e_1 AND e_2 AND GRID SEARCHED e_1 AND e_2 , USING GAUSS2, FOR P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC.

FIGURE 6.16 NORMALIZED NOISE TRANSFERS USING PREDICTED AND GRID SEARCHED P2Cs OF $G_4(s)$ WITH $\zeta=0.5$ AND $T=1$ SEC. USING GAUSS2

As for $G_2(s)$, the solution was not obtainable by use of the calculation algorithm for some values. When $\omega_n \leq 2$, the predicted values of e_1 and e_2 were calculated by iterating until the minimum value of NCE_2 was found. For values of $\omega_n > 2$, no such difficulty existed.

FIG. 6.13 shows the predicted values of e_1 and e_2 . The normalized noise transfers obtained using the predicted PlCs and P2Cs when subjected to GAUSS2 are compared in FIG. 6.14. A comparison of the $D(z)$ error terms for the predicted and grid searched P2Cs is given in FIG. 6.15. The final graph, FIG. 6.16 compares the normalized noise transfers obtained using the predicted and grid searched P2Cs using GAUSS2.

The improvement in the normalized noise transfers when using the predicted P2Cs instead of the predicted PlCs varies from essentially zero at $\omega_n \approx 1.5$ to approximately 13% at $\omega_n = 0.25$. For $\omega_n > 10.0$, the improvement is greater than 10%.

The differences in e_1 are -1% at $\omega_n = 0.25$ and -12% at $\omega_n = 64.0$. The differences for e_2 are -5.9% and + 11% respectively as calculated from FIG. 6.15. From FIG. 6.16, the maximum noise transfer difference is -14.7% at $\omega_n = 0.5$ and the minimum difference is zero for $\omega_n \approx 1.25$. For $\omega_n > 4$, the difference is no greater than -3%.

6-6 SOME CONCLUSIONS FOR THE P2Cs

The reduction of the system noise transfer varied from 25% to 70% when the predicted P2Cs replaced the minimum time deadbeat controllers for any of the plants tested. These percentages were calculated using GAUSS2 as the noise source.

The largest improvements occurred for $G_2(s)$ and $G_4(s)$ which had five term P2Cs. This represented a 2/3 increase in the response time when compared to the minimum time deadbeat controller. The ranges of improvements were 40% to 66% for $G_2(s)$ and 25% to 70% for $G_4(s)$.

When the larger noise transfer reductions are considered, the trade-off of response time would generally be justified. However for some regions, the numerical value of the percentage noise reduction achieved was much less than the numerical value of the percentage increase in the response time. In these regions, the decision must be based on the importance of the absolute noise reduction.

For $G_1(s)$ and $G_3(s)$, the increase in the response time was 100%. The ranges of the noise reductions were 35% to 45% for $G_1(s)$ and 40% to 55% for $G_3(s)$. In both cases, the numerical value of the percentage noise reduction was much less than the numerical value of the percentage increase in the response time. Therefore the decision to use these P2Cs must be based on the importance of the absolute noise reduction.

Again for the predicted P2Cs, a transition region

in the noise transfer curves existed for the same values of τ as found for the PlCs. The shape of the noise transfer curves using the predicted P2Cs was virtually the same as for the corresponding predicted PlCs for all plants tested. The absolute values of the noise reduction were the only differences. The reasons for the shape of these curves are the same as those given in Section 5-7.

The agreement between the predicted and "grid searched" P2Cs was not as good as the agreement obtained for the PlCs. Generally the agreement between the e_1 values was better than the agreement between the e_2 values.

The maximum difference for the noise transfers was 20% but the difference generally was $< 10\%$. For $\tau < 0.5T$, the differences were the largest. These differences dropped to $< 5\%$ for $\tau > T$ for all plants tested.

The differences between the e_1 values were generally comparable to the noise transfer differences. However, the differences between the e_2 values were generally numerically larger than the noise transfer differences.

Since GAUSS2 is not a pure white noise source, it has some correlation with the step input. The P2C acts on the system for a longer period of time than does the corresponding PlC. Therefore any differences due to the non-white noise source would be magnified by the P2C. This is the reason that the differences between the predicted and "grid searched" P2Cs are larger than for the PlCs.

For large values of τ , the system response occurs mostly in the first part of the controller cycle. Therefore changes in the later terms of the controller have a lesser effect than when τ is small. This accounts for the better agreement between the "grid searched" and predicted P2Cs for larger values of τ . For $\tau > T$, the predicted P2Cs appear to be an acceptable approximation of the minimum noise transfer controllers for the non-white noise sources used. For $\tau < T$, only $G_2(s)$ exhibits any continuing agreement between the predicted and "grid searched" P2Cs. The agreement for the P2Cs is never as good as the agreement for the PlCs for any of the plants tested.

Again there were absolute maximum and absolute minimum noise reductions in the noise transfer curve for all plants tested. The same discussion used in Section 5-7 may be applied in this case.

CHAPTER VII

OTHER EXTENDED CONTROLLERS

In the first section, the predicted controllers extended by 3 terms for $G_1(s)$ are presented. No grid search calculations are presented for this case. This controller is abbreviated as P3C which stands for Plus Three Extra Terms Controller.

In the second section, some predicted controllers using a reduced sampling period are studied. Some results for the straight line approximations to the predicted PLCs are presented in the third section.

In each section, some conclusions are presented for the particular situation under consideration.

7-1 PREDICTED P3Cs OF $G_1(s)$

For these calculations, $T = 1$ second was the sampling rate and α was varied from 0.25 to 64.0.

Adding three extra terms to the controller results in three degrees of freedom in its calculation. This conclusion is reached in the same way as for the PLCs and P2Cs. These three degrees of freedom may be represented by e_1 , e_2 and e_3 in the denominator of the controller.

To calculate the predicted P3Cs, the expression for NCE when $n = 5$ was written from (3.19). Using a method

analogous to that used in Chapters 5 and 6, the coefficients h_0, h_1, h_2, h_3, h_4 and e_4 were written as functions of e_1, e_2 and e_3 . Using these expressions, (3.19) was rewritten as a function of e_1, e_2 and e_3 .

By iterating e_3, e_2 and e_1 in that order a grid search for the global minimum value of NCE at each value of α was performed. This approach was used rather than the calculation algorithm used in Chapters 5 and 6 as only a few results were taken. The programming complexity associated with the algorithm was thus replaced by a simpler iteration process. The values obtained for e_1, e_2 and e_3 are given only to 2 significant figures. Once the predicted values were known, the resultant controllers were tested using the GAUSS2 noise input.

The predicted values of e_1, e_2 and e_3 are plotted in FIG. 7.1. A comparison of the normalized noise transfers of the systems containing the predicted P1Cs, P2Cs and P3Cs is shown in FIG. 7.2. GAUSS2 was the noise source used in all cases.

Only the predicted P3Cs were calculated for this plant as these would indicate the improvement obtained when the P2Cs are replaced by the P3Cs. Also a grid search using GAUSS2 is very costly in terms of computing time and therefore was not performed.

The improvements in the normalized noise transfer when using the P2Cs rather than the P1Cs varied from 6% to 13% as taken from FIG. 7.2. The corresponding improvements

when the P3Cs replace the P2Cs are 0.5% to 6%. These percentages are measured with respect to the corresponding minimum time controller.

For most situations, the advantages of the lowered noise transfers obtained when using the P3C are outweighed by the increased complexity, slow response time to a stop input and the cost of realization of the P3Cs. Therefore in general the P3Cs is not useful.

However for some particular situations, where long term noise performance is the most important requirement, the use of these P3Cs may be warranted. From FIG. 7.2, the noise improvement is $> 3\%$ for $\alpha > 1.0$. This is the region where the largest noise improvement occurs as compared to the P2C.

7-2 PLCs WITH A REDUCED SAMPLING PERIOD

The extended controllers considered so far have used the same sampling rate as the minimum time deadbeat controller. The case will now be considered where the sampling rate used with the PLC is such that the total time to deadbeat for both the minimum time controller and the PLC is the same.

The total response time of the minimum controller is nT . Therefore the sampling period of the PLC must be reduced to $(\frac{n}{n+1})T$, where n is the order of the minimum controller. In this way, the total response time to deadbeat is constant for both controller types.

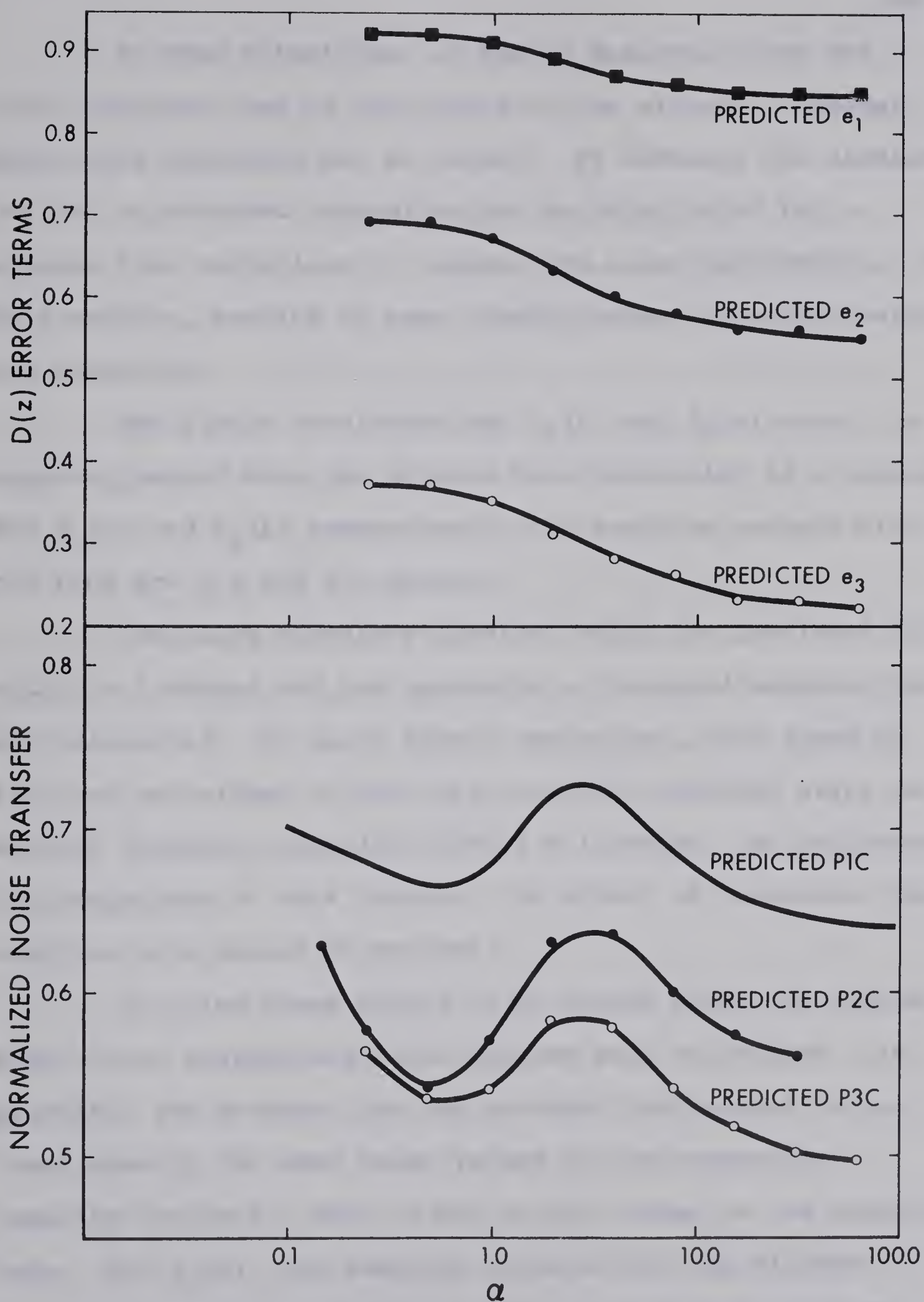


FIGURE 7.1 PREDICTED e_1 , e_2 and e_3 FOR P3Cs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 7.2 NORMALIZED NOISE TRANSFERS USING PREDICTED P1Cs, P2Cs AND P3Cs OF $G_1(s)$ WITH $T=1$ SEC. USING GAUSS2

In some situations, it may be desirable that the total response time of the system not be altered. However some noise reduction may be needed. By reducing the sampling period, an extended controller can be substituted for a minimum time controller to improve the noise performance. In this section, results of some investigations into this method are presented.

The plants considered are $G_1(s)$ and $G_2(s)$ where the sampling period with the minimum time controller is 1 second. For $G_1(s)$ and $G_2(s)$ respectively, the sampling periods with the PLCs are $2/3$ and $3/4$ second.

The noise transfers obtained using the predicted PLCs with $T = 1$ second and the appropriate increased sampling rate are calculated. To allow direct comparison, both types of PLCs are normalized to the noise transfer obtained using the minimum deadbeat controller with $T = 1$ second. By performing the comparison in this fashion, the effect of increasing the sampling rate should be evident.

To allow these plants to be tested using the digital simulation, appropriate noise sources must be defined. In practice, the minimum time and extended controllers do not "see" exactly the same noise values at the respective sampling instants. This is due to the change in the sampling rate. For $G_1(s)$, the sampling instants for the minimum time and extended controllers will only coincide every $2T$ periods and similarly every $3T$ periods for $G_2(s)$.

A sequence of random numbers is generated from which the appropriate noises sources are taken. The total sequence generated is GAUSS2A.

For $G_1(s)$, a random number was generated every $1/3$ second in the simulation. Therefore every third number was chosen to simulate the noise source when $T = 1$ second. Similarly every second number was chosen to simulate the noise source when $T = 2/3$ second. The experimental run in both cases was 256 seconds. The results for the predicted PLCs using both sampling rates are shown in FIG. 7.3.

For $G_2(s)$, a random number was generated every $1/4$ second in the simulation. When $T = 1$ second, every fourth number was chosen. When $T = 3/4$ second, every third number was chosen. Again the length of the experimental run was 256 seconds. These results are shown in FIG. 7.4.

The noise sources used in these simulations were called GAUSS2AM.

From FIG. 7.3 and FIG. 7.4, the noise reduction obtained when using the increased sampling rate predicted PLCs is not as great as that obtained when using the normal predicted PLCs. For both plants, this difference is largest for small values of α . The results for the two types of PLCs agree more closely as α increases. For $\alpha > 3$ the increased sampling rate predicted PLC is slightly better for $G_1(s)$.

A given plant requires that the controllers have higher equivalent gains during the sampling period when the

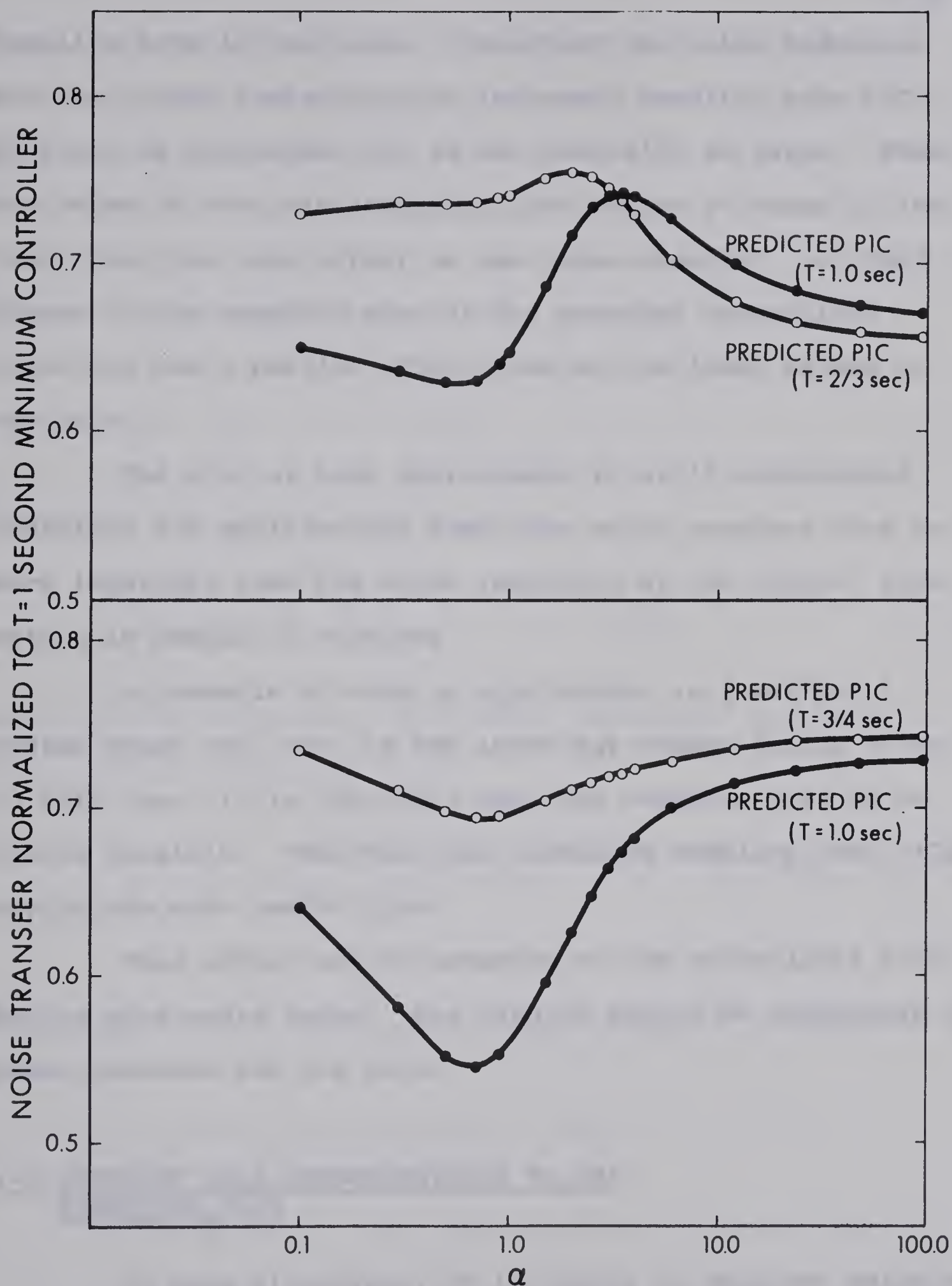


FIGURE 7.3 NORMALIZED NOISE TRANSFERS USING PREDICTED PICs OF $G_1(s)$ WITH $T=2/3$ & 1 SEC. USING GAUSS2AM

FIGURE 7.4 NORMALIZED NOISE TRANSFERS USING PREDICTED PICs OF $G_2(s)$ WITH $\beta=1$ AND $T=3/4$ & 1 SEC. USING GAUSS2AM

sampling rate is increased. Therefore the noise reduction for the system containing the increased sampling rate PLC in place of the normal PLC is not generally as large. When the value of the pole increases, the number of terms in the controller has less effect on the noise transfer. A small change in the sampling rate of the extended controllers therefore has a smaller effect than at the lower values of the pole.

The size of this improvement is still substantial. Therefore for applications where the total response time is more important than the noise reduction of the system, this method is useful to consider.

An example of such an application is a set-point system where the level of the input may change fairly often. In this case, it is important that the response time be as low as possible. Therefore the increased sampling rate PLCs may be the more useful type.

This method may be extended to the controllers with two or more extra terms. The results should be comparable to those obtained for the PLCs.

7-3 STRAIGHT LINE APPROXIMATIONS TO THE PREDICTED PLCs

In some situations, it is useful to consider methods to approximate the predicted PLCs. By using an approximation, a more rapid method of calculation is available.

In Chapter 3, a "straight line" approximation is mentioned. This method was found to be useful for particular plants. The approximation minimizes the differences between the denominator terms of the controller and a straight line from 0 at $t = 0$ to 1.0 at $t = nT$. The values of the straight line at the sampling instants are:

$$\left(\frac{1}{n}\right), \left(\frac{2}{n}\right), \dots, \left(\frac{n-1}{n}\right) \quad (7.1)$$

Therefore the "ideal" errors are:

$$\left(1 - \frac{1}{n}\right), \left(1 - \frac{2}{n}\right), \dots, \left(1 - \frac{n-1}{n}\right) \quad (7.1a)$$

The differences between the "ideal" errors and the error terms of the controller are:

$$\left(1 - \frac{1}{n} - e_1\right), \left(1 - \frac{2}{n} - e_2\right), \dots, \left(1 - \frac{n-1}{n} - e_n\right) \quad (7.2)$$

Using the (7.2), the following equation may be written:

$$\underline{\text{Straight Line Difference}} \equiv \text{SLD} =$$

$$\left(1 - \frac{1}{n} - e_1\right)^2 + \left(1 - \frac{2}{n} - e_2\right)^2 + \dots + \left(1 - \frac{n-1}{n} - e_n\right)^2 \quad (7.3)$$

For each plant, the error terms of the controller e_2, e_3, \dots, e_n may be expressed in terms of e_1 . Therefore (7.3) may be rewritten as:

$$\text{SLD} = f(e_1) \quad (7.4)$$

The value of e_1 which yields the absolute minimum value of SLD may be found from (7.4) by simple differentiation as in Chapter 5. Taking the derivative of (7.4) with respect to e_1 and setting the result to zero yields:

$$\frac{d}{de_1} [\text{SLD}] = \frac{d}{de_1} [f(e_1)] = 0 \quad (7.5)$$

The roots of (7.5) correspond to the local maxima and minima of SLD. From these roots, the value of e_1 which satisfies the deadbeat requirements and corresponds to the minimum of SLD is chosen.

The other coefficients of the "straight line" approximation PLC are then calculated using this value of e_1 . In FIG. 7.5 and FIG. 7.6, the predicted values of e_1 are compared to the "straight line" approximations of e_1 for $G_1(s)$ and $G_2(s)$ respectively.

From FIG. 7.5 and FIG. 7.6, it is apparent that the approximation is most accurate for $\alpha < 1$. For $\alpha > 1$, the approximation is rather inaccurate and therefore not useful. Although the results are not shown, the approximation is also useful for $\omega_n < 1.0$ when the PLC of $G_4(s)$ is used.

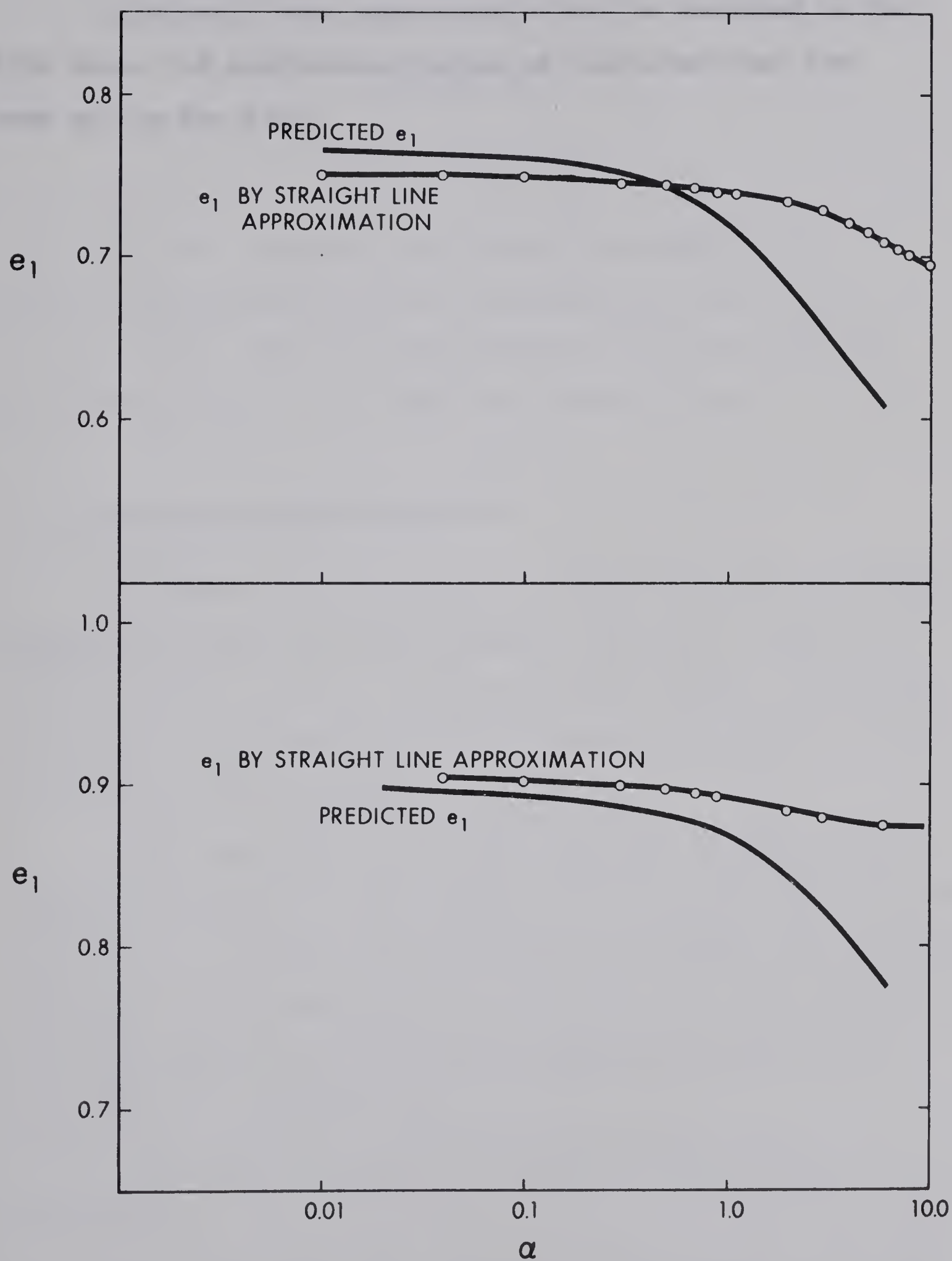


FIGURE 7.5 PREDICTED e_1 AND STRAIGHT LINE APPROXIMATION e_1 FOR PLCs OF $G_1(s)$ WITH $T=1$ SEC.

FIGURE 7.6 PREDICTED e_1 AND STRAIGHT LINE APPROXIMATION e_1 FOR PLCs OF $G_2(s)$ WITH $\beta=1$ AND $T=1$ SEC.

Similarly this approximation may be extended to the P2Cs where the approximate ranges of usefulness are the same as for the PlCs.

CHAPTER VIII

GENERAL CONCLUSIONS

In this chapter, some general conclusions and observations based on the work performed for this thesis are presented. Then some extensions of this work and the use of these results in other ways are discussed.

8-1 CONCLUSIONS AND OBSERVATIONS

A design criterion for the calculation of non-minimum deadbeat controllers which minimize the noise transfer of linear systems has been presented in this thesis. The noise is assumed to be white as this allows the application of the criterion to be as practical as possible.

This approach to the noise problem is novel as the additional constraint of deadbeat is maintained while the noise transfer is minimized and the criterion excludes any explicit reference to the input noise. In Section 3-5, the physical reasoning behind the criterion is presented as part of the derivation of the criterion.

Two types of extended controllers have been considered:

1. Controllers using the same sampling rate as the minimum time deadbeat controllers.
2. Controllers using a higher sampling rate such

that the response time is the same as for the minimum time deadbeat controllers.

For the first case, controllers with up to three extra terms were considered. As stated in Section 5-7, the noise reduction was substantial when the predicted PlC was used in place of the minimum time controller. From Sections 6-6 and 7-1, the extra noise reductions obtained by using the predicted P2C or P3C in place of the predicted PlC were not as large as the initial reduction.

Therefore when the possible noise reduction is weighed against the disadvantages of the increased response time and extra controller complexity, the optimum practical extension is one term. However for applications where the absolute noise reduction is the most important consideration, the longer extensions will be useful.

For the second case, only the PlCs for $G_1(s)$ and $G_2(s)$ were considered. The noise reductions obtained using these controllers were not as large as those for the first type. However as stated in Section 7-2, the reductions were still substantial. Similar results should be obtainable with other plants. A similar relationship should also exist between the two types of extended controllers when other extensions are considered. This type of extended controller is most useful where the response time is the most important consideration.

The PlCs predicted by the design criterion are a good approximation of the minimum noise transfer controllers for

various non-white noise sources. This was verified for all plants tested. For the P2Cs predicted by the design criterion, this approximation was generally not as good as for the PLCs.

It was also observed that some predicted PLCs had a lower error to a ramp or an acceleration input than did the corresponding minimum time deadbeat controllers.

Although only four types of plants were tested, the conclusions reached should apply to any linear plant in a similar system.

8-2 EXTENSIONS AND DISCUSSION

For all systems tested, the noise was injected before the controller. If the noise is injected between the plant and the controller, the extended controllers with an unchanged sampling rate should cause only a slight reduction in the noise transfer when compared to the noise transfer obtained using the minimum time controller. Most of the noise will be superimposed directly on the output due to the slow controller response time. The extended controllers with the increased sampling rate may cause slightly more improvement because:

1. The response time is the same as for the minimum time deadbeat.
2. The loop gains for the extended controllers are lower than for the minimum time controllers.
3. The extended controllers can sense changes at the

output more quickly, due to the higher sampling rate, than can the minimum time controllers.

Although the calculation of a design criterion would be more complicated, the method should be applicable to extended deadbeat controllers designed for ramp or acceleration inputs when a linear plant is used. It should still be possible to minimize the "weighting sequence" of an extended deadbeat controller for these two inputs.

This design criterion should also be applicable to systems containing a linear plant and a linear non-unity feedback branch.

If an otherwise linear system contains a continuous saturable type of non-linearity, the design criterion may be of some use. By extending the controller, the gains are lowered and thus the system may operate in the linear part of the non-linearity during a greater portion of the time. As long as the non-linearity operates in its linear region, the noise transfer of the system will be reduced. For non-linearities which do not contain linear regions, this design criterion would not be useful.

BIBLIOGRAPHY

1. J. T. TOU, "Statistical Design of Digital Control Systems," I. R. E. TRANSACTIONS ON AUTOMATIC CONTROL, vol. AC-5, September 1960, pp. 290-297.
2. VON J. ACKERMANN, "Anwendung der Wiener-Filtertheorie zum Entwurf von Abtastreglern mit beschränkter Stelleistung," REGELUNGSTECHNIK, HEFT 8, January 1968, pp. 353-359.
3. R. C. MARTELL, "Deadbeat Response by Digital Controllers," M.Sc. Thesis, University of Alberta, 1967.
4. B. C. KUO, "Analysis and Synthesis of Sampled-Data Control Systems," Prentice Hall, 1963.
5. JOHN R. RAGAZZINI and GENE F. FRANKLIN, "Sampled-Data Control Systems," McGraw-Hill, New York, 1958
6. JOHN B. THOMAS, "An Introduction to Statistical Communication Theory," John Wiley & Sons, Inc., 1969.
7. J. S. BENDAT and A. G. PIERSON, "Measurement and Analysis of Random Data," John Wiley & Sons, Inc., 1966.
8. J. T. TOU, "Statistical Design of Linear Discrete-Data Control Systems via the Modified z - Transform Method," J. Franklin Inst., vol. 271, no. 4, April 1961, pp. 249-262.

9. J. T. TOU and K. S. PRASANNA KUMAR, "Statistical Design of Discrete-Data Control Systems Subject to Power Limitations," J. Franklin Inst., vol. 272, Sept. 1961, pp. 171-183.
10. D. P. LINDORFF, "Discussion of (1)," IRE Trans. on Auto. Control, vol. AC-5, Sept. 1960, pp. 296-297.
11. K. STEIGLITZ, P. A. FRANESZEK, A. H. HADDAD, "Discussion of (1)," IEEE Trans. on Auto. Control, vol. AC-10, April 1965, pp. 216-217.
12. E. I. JURY, "Comments on the Statistical Design of Linear Sampled-Data Feedback Systems," IEEE Trans. on Auto. Control, vol. AC-10, April 1965, pp. 215-216.
13. F. CSAKI, "Discussion of (1)," IEEE Trans. on Auto. Control, vol. AC-11, Jan. 1966, pp. 149-150.
14. S.S.L. CHANG, "Statistical Design Theory for Digital-Controlled Continuous Systems," Trans. AIEE, vol. 77, pt. 2, Sept. 1958, pp. 191-201.

APPENDIX A

HYBRID COMPUTER SIMULATION

The methods used to simulate the plant are standard analog techniques and therefore will not be illustrated in this appendix. This appendix contains only the details of the program written to realize the digital controller.

A-1 DIGITAL CONTROLLER REALIZATION

The digital program in the hybrid simulation consisted of three sections which

1. Realized the controller
2. Realized the zero-order hold
3. Calculated the noise transfer of the system at the end of the test run.

The realization of the controller function was hybrid. The coefficient h_0 was realized on the analog computer while all other coefficients appear only in the digital program. A digital to analog converter acted as the zero-order hold. The diagram of the realization of the controller and zero-order hold is shown in FIG. A.1.

The controller was realized using the Direct Digital Programming discussed in Chapter 2. It was shown that h_0 operates on the present input sample to the controller. All

other coefficients of the $D(z)$ operate only on the delayed information.

Therefore h_0 was realized on the analog computer to make the length of the sampling window as short as possible for the equipment used. The actual length of the sampling window was 40 microseconds in the hybrid realization. As shown in FIG. A.1, $h_0 BE_1(t)$ was summed with the resultant products of the delayed information and the other controller coefficients.

The multiplications of the controller coefficients and the delayed inputs and outputs were carried out between samples. The resulting values were then summed and the resultant $BE_3(nT)$ was D/A converted. Then at the sampling instants, the instantaneous value of $h_0 BE_1(t)$ was summed with $BE_3(nT)$. The resultant was A/D converted and then D/A converted immediately. Both the preceding and following samplers shown with the controller in FIG. 1.1 were simulated by the A/D conversion. The zero-order hold was simulated by the D/A conversion. The length of the sampling window was thus set by this A/D and D/A sequence.

To relate the FOCAL program given in Section A-2 to FIG. A.1, the following definitions are given:

1. $N_1 = n-1$ where n = the order of the controller
2. $X(i) = -BE_1(t-iT); i = 1, 2, \dots, n$
3. $Y(i) = BE_2(t-iT); i = 1, 2, \dots, n$
4. $H(i) = h_i; i = 1, 2, \dots, n$

5. $E(i) = e_1 ; i = 1, 2, . . . , n$
6. $z = BE3(nT)$

The definitions of the commands used in FOCAL are:

1. S \equiv SET
2. T \equiv TYPE
3. % 4.01 \equiv FORMAT STATEMENT (SAME AS
F 4.1 IN FORTRAN)
4. FSQT \equiv SQUARE ROOT
5. FABS \equiv ABSOLUTE VALUE

The command FNEW is used in FOCAL to perform a variety of functions. These are A/D conversions, D/A conversions and the control of the operating state of the analog computer integrators.

An A/D conversion is specified, for example, by

```
S Y(1)=FNEW(1,4,0)
```

which means A/D convert the signal on A/D channel 4 and set the converted value equal to Y(1). The 0 is a dummy variable and 1 defines an A/D conversion.

A D/A conversion is specified, for example, by

```
S A=FNEW(2,3,Y(1))
```

which means D/A convert the value Y(1) onto D/A channel 3.

The 2 specifies a D/A conversion and A is a dummy variable.

The command

```
S WAIT=FNEW(5,0,0,)
```

is used to synchronize the program with the clock of the digital computer. This clock sets the length of the sampling period and also the start of the experimental run. When the digital computer is in a wait loop, the only way the program can proceed to the next command is by the reception of a clock pulse which occurs at the sampling instants.

The FNEW commands which control the operating mode of the analog integrators are self-explanatory.

The multiplications of the controller coefficients and the delayed inputs and outputs and the summing of the resultant values are performed by statement 06.10.

Statement 06.10 performs both the multiplications and the summations which yields $BE3(nT)$. The shifting and the time delaying of the input and output values are performed by statements 06.50 to 06.80.

When the analog values are A/D converted in statements 07.15 and 07.20 to allow the calculation of the input and output root-mean-square values, the divisor 2047 is used. This divisor is the octal value in the A/D converter which corresponds to 10 volts on the analog computer. This division is necessary to yield the printed values of NI and

NO in volts. These are the root-mean-square values of the input and output noise respectively.



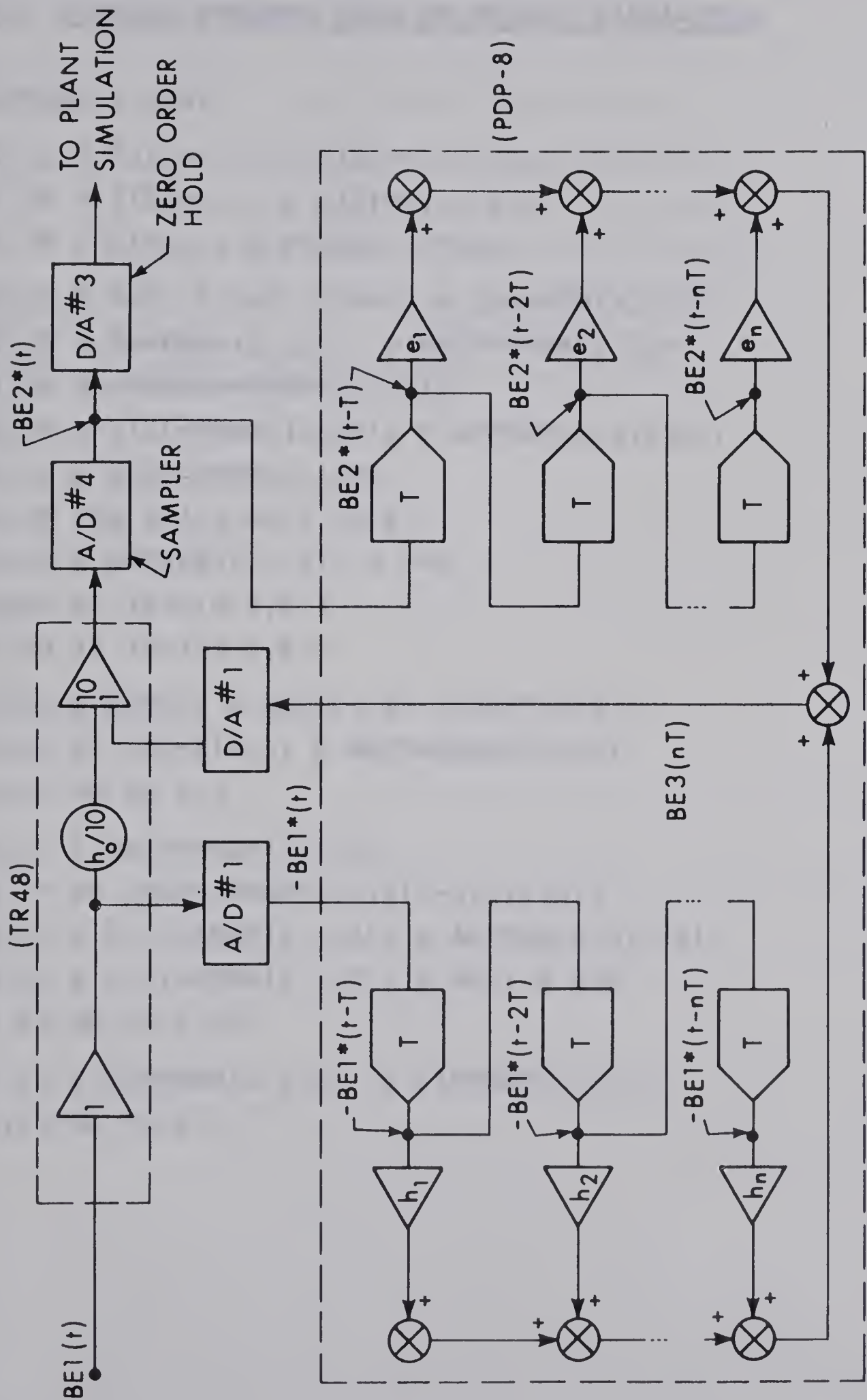


FIGURE A.1 HYBRID REALIZATION OF THE DIGITAL CONTROLLER

A-2 DIGITAL PROGRAM USED IN HYBRID SIMULATION

C-FOCAL., 1968

```

01.10 S H(1)=...; S H(2)=...; ...
01.20 S E(1)=..., S E(2)=...; ...
01.70 S N1=...; S P=100; S T=1

02.10 S M=0; S Z=0; S K=0; S E1=FNEW(2,1,0)
02.15 S E3=FNEW(2,3,0); S WAIT=FNEW(5,0,0)
02.20 S OPERATE=FNEW(4,10,1)
02.30 S Y(1)=FNEW(1,4,0); S A=FNEW(2,3,Y(1))
02.32 S X(1)=FNEW(1,1,0)
02.35 FOR R=1,1,K+1; DO 6.1
02.40 S G=FNEW(2,1,Z); S Z=0
02.45 IF (K-1)6.8,6.7
02.50 IF (K-3)6.6,6.5

03.10 S M=M+1; S K=K+1; IF (P-M)7.1,7.1
03.20 IF (N1-K)4.1; S WAIT=FNEW(5,0,0)
03.25 GO TO 2.3

04.10 S WAIT=FNEW(5,0,0)
04.20 IF (FABS(FNEW(1,1,0))-5)5.1,5.1
04.30 S Y(1)=FNEW(1,4,0); S A=FNEW(2,3,Y(1))
04.35 S X(1)=FNEW(1,1,0); S K=0; S Z=0
04.40 GO TO 2.35

05.10 S E3=FNEW(2,3,0); S E1=FNEW(2,1,0)
05.20 GO TO 3.1

```



```

06.10 S B=H(R)*X(R)+E(R)*Y(R)+Z; S Z=B
06.50 S X(5)=X(4); S Y(5)=Y(4)
06.60 S X(4)=X(3); S Y(4)=Y(3)
06.70 S X(3)=X(2); S Y(3)=Y(2)
06.80 S X(2)=X(1); S Y(2)=Y(1); GO TO 3.1

07.10 S WAIT=FNEW(5,0,0); S HOLD=FNEW(4,10,4)
07.15 S NI=FSQT(10*(1/M*T)*(FNEW(1,7,0)/2047))
07.20 S NO=FSQT(10*(1/M*T)*(FNEW(1,8,0)/2047))
07.25 T %12.06, " IN ",NI," OUT ",NO,!!
07.30 T %12.06, " % ",(NO*100)/N1,!!!
07.35 S IC(INITIAL CONDITION)=FNEW(4,10,2)

08.10 QUIT

```


APPENDIX B

DIGITAL COMPUTER SIMULATION OF THE DIGITAL CONTROLLERS

The digital computer program used to simulate the deadbeat controllers is presented in this appendix. The program sections which contain the variables for the state equations given in Chapter 1 and the calculation algorithms for the minimum time and extended deadbeat controllers are not listed. These sections are easily programmed directly from the lists of equations given in the text of the thesis.

B-1 DERIVATION OF PROGRAM

This program incorporates three sections which

1. Verify the deadbeat response of the system.
2. Subject the controllers to the noise sources defined in Chapter 4.
3. Calculate the noise transfer of the system containing the minimum time deadbeat controller and the normalized noise transfers of the extended deadbeat controllers.

These sections are numbered in the program listing.

The systems are realized by combining the discrete state equations with the direct digital realization of the

digital controller. In this program, $AH(N)$ and $AE(N)$ are used to represent h_n and e_n respectively.

B-2 DIGITAL PROGRAM

```

100  CONTINUE
C    AIN2 AND AOUT2 ARE USED TO CALCULATE
C    INPUT AND OUTPUT MEAN-SQUARE NOISE
    AIN2=0.0
    AOUT2=0.0
C    SET X1(1) TO XN(1) TO ZERO
    X1(1)=0.0
      :
    XN(1)=0.0
C    INITIALIZE OTHER COEFFICIENTS
    ICYCLE=N
    K=1
    JJ=1
    COUNT=1.0
    XOUT1(1)=0.0

C    CONTROLLER REALIZATION BY DIRECT DIGITAL
C    PROGRAMMING
C    CALCULATE INPUT TO CONTROLLER,BE1(JJ)
101  BE1(JJ)=1.0-XOUT1(JJ)
    BEA1=0.0
    BEB1=0.0
    DO 560 JK=1, K
      BEA(JK)=AH(JK)*BE1(JJ-JK+1)
      BEA1=BEA1+BEA(JK)
560  CONTINUE
    IF(K.LT.2) GO TO 562
    DO 561 JL=2,K
      BEB(JL)=AE(JL)*BE2(JJ-JL+1)
      BEB1=BEB1+BEB(JL)

```



```

561  CONTINUE
C    CALCULATE OUTPUT OF CONTROLLER, BE2(JJ)
562  BE2(JJ)=BEA1+BEB1
C    STATE EQNS. TO CHECK DEADBEAT (SECTION 1)
    X1(K+1)=. . .
        :
    XN(K+1)=. . .
    JJ=JJ+1
    XOUT1(JJ)=X1(K+1)
    IF (COUNT .GE.7.0) GO TO105
    COUNT=COUNT+1.0
    IF(K.GE.ICYCLE) GO TO102
    K=K+1
    GO TO101
102  X1(1)=K1(K+1)
        :
    XN(1)=XN(K+1)
    K=1
    GO TO101
105  X1(1)=0.0
        :
    XN(1)=0.0
    K=1
    J=1
    XOUT(1)=0.0
C    CALCULATE INPUT TO CONTROLLER, BE1(J)
C    WHERE YFL1(J) IS THE NOISE SOURCE VALUE
111  BE1(J)=YFL1(J)-XOUT(J)
    BEB1=0.0
    BEA1=0.0
    DO 550 JM=1,K
    BEA(JM)=AH(JM)*BE1(J-JM+1)
    BEA1=BEA1+BEA(JM)
550  CONTINUE
    IF (K.LT.2) GO TO552

```



```

DO 551 JN=2,K
  BEB(JN)=AE(JN)*BE2(J-JN+1)
  BEB1=BEB1+BEB(JN)
551 CONTINUE
C   CALCULATE OUTPUT OF CONTROLLER, BE2(J)
552 BE2(J)=BEA1-BEB1
C   STATE EQNS. TO TEST SYSTEM IN
C   PRESENCE OF NOISE INPUT (SECTION 2)
  X1(K+1)=. . .
  :
  XN(K+1)=. . .
  J=J+1
  XOUT(J)=X1(K+1)
C   NRUN=256 OR 1024, ICYCLE=N
  IF(J.GE.(NRUN+1)) GO TO113
  IF(K.GE.ICYCLE) GO TO112
  K=K+1
  GO TO111
112 X1(1)=X1(K+1)
  :
  XN(1)=XN(K+1)
  K=1
  GO TO111
C   CALCULATE THE SUM OF SQUARES OF THE
C   NOISE IN AND OUT (SECTION 3)
113 DO 114 I=1, NRUN
  AIN2=AIN2+YFL1(I)**2
  AOUT2=AOUT2+(XOUT(I+1))**2
114 CONTINUE
C   CALCULATE MEAN-SQUARE TRANSFER RATIO
  RMSNSE=AOUT2/AIN2
C   IF CHECK=0.0, RETURN TO MINIMUM TIME
C   CONTROLLER PROGRAM
  IF(CHECK.EQ.0.0) GO TO150
C   CALCULATE NORMALIZED NOISE TRANSFER

```



```
C      FOR EXTENDED CONTROLLER WHERE RMSDBT
C      IS THE NOISE TRANSFER OF MINIMUM
C      CONTROLLER
      RMS=RMSNSE/RMSDBT
C      RETURN TO EXTENDED CONTROLLER PROGRAM
      GO TO 170
```


B30005